EUDAS

Anwender-
Datenverwaltungssystem

VERSION 3

BENUTZERHANDBUCH

Ausgabe September 85

Dieses Handbuch und das zugehörige Programm sind urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere der Vervielfältigung in irgendeiner Form, bleiben dem Autor vorbehalten.

Es kann keine Garantie dafür übernommen werden, daß das Programm für eine bestimmte Anwendung geeignet ist. Die Verantwortung dafür liegt beim Kunden.

Das Handbuch wurde mit größter Sorgfalt erstellt. Für die Korrektheit und Vollständigkeit der Angaben wird aber keine Gewähr übernommen. Das Handbuch kann jederzeit ohne Ankündigung geändert werden.

(c) Thomas Berlage 1985

Vorwort

Lieber EUDAS-Benutzer !

Dieses Handbuch soll Sie bei Ihrer Arbeit mit EUDAS begleiten. Obwohl EUDAS nicht schwierig zu bedienen ist, gibt es doch eine Reihe von Dingen zu lernen, ehe Sie ein EUDAS-Spezialist geworden sind.

Um Ihnen diesen Weg möglichst einfach zu machen, ist das Handbuch in vier Teile unterteilt. Die ersten beiden Teile bilden eine Einführung, der Rest ist zum Nachschlagen gedacht.

Der erste Teil des Handbuchs beschäftigt sich mit den grundlegenden Konzepten von EUDAS. Diesen Teil sollten Sie auf jeden Fall lesen. Dort werden Sie anhand von einfachen Beispielen in die Benutzung von EUDAS eingeführt. Anschließend sollten Sie so weit sein, daß Sie EUDAS für eigene Zwecke verwenden können.

Mit Hilfe von Beispielen werden Ihnen die wichtigsten Möglichkeiten von EUDAS vorgeführt. Sie sollten diese Beispiele des ersten Teils selbst am Rechner nachvollziehen, um ein Gefühl für die Bedienung zu bekommen. Nur so können Sie in kürzester Zeit die wesentlichen Funktionen kennenlernen.

Wenn Ihre Ansprüche dann wachsen, sollten Sie sich mit dem zweiten Teil befassen. Hier erhalten Sie Einblick in weitergehende Möglichkeiten von EUDAS. Die einzelnen Kapitel sind relativ unabhängig voneinander, so daß Sie nur die für Sie interessanten genauer durchlesen müssen.

Der dritte Teil des Handbuchs dient dann zum Nachschlagen der einzelnen Möglichkeiten. Dort sind alle Daten kompakt und (hoffentlich) klar zusammengefaßt. Das bedeutet aber auch gleichzeitig, daß Sie diesen Teil erst dann ganz verstehen können, wenn Sie die vorherige Einführung gelesen haben.

Der letzte Teil schließlich beschäftigt sich mit den Möglichkeiten für ELAN-Programmierer. Obwohl Sie einige Teile auch schon mit wenigen Kenntnissen benutzen können, sollten Sie sich erst dann daran wagen, wenn Sie wirklich sicher sind, daß Sie ELAN verstehen.

Trotz größter Bemühungen kann das Handbuch natürlich nicht frei von Unklarheiten und Fehlern sein. Anregungen und Kritik sind daher dringend erwünscht, um diese Dokumentation zu verbessern. Insbesondere Hinweise auf Stellen, die Ihnen besondere Schwierigkeiten gemacht haben, können dazu dienen, auch EUDAS selbst zu überarbeiten. Scheuen Sie sich daher nicht, auch unwesentliche Kritikpunkte anzubringen.

Und nun viel Spaß bei Ihrer Arbeit mit EUDAS !

INHALT

Vorwort . . i
Inhalt . . . iii

I. EINFÜHRUNG IN DIE BENUTZUNG

1 Was kann EUDAS ?

1.1 Textverarbeitung und Datenverwaltung . 1
1.2 EUDAS als Karteikasten . . 3
1.3 Drucken 5
1.4 Grenzen 6

2 Installation

2.1 Lieferumfang 7
2.2 Single-User . . . 8
2.3 Multi-User 8

3 Bedienung der Menüs

3.1 Menüprinzipien . . 11
3.2 Hilfestellung 13
3.3 Struktur der Menüs . . . 13
3.4 Zusammenfassung . . 14

4 EUDAS-Dateien

4.1 Textdateien und EUDAS-Dateien . 15
4.2 Aufbau von EUDAS-Dateien . 16
4.3 Archiv . . . 17
4.4 Öffnen . 19

5 Ansehen und Suchen

5.1 Bildschirmaufbau . 21
5.2 Bewegen 22
5.3 Suchen 23
5.4 Suchbedingungen 25

6 Einrichten und Ändern

6.1 Neue Datei einrichten . . . 27
6.2 Ändern und Einfügen 28
6.3 Satzeditor . . 29
6.4 Sichern . 31

7 Drucken mit EUDAS

7.1 Druckmuster 33
7.2 Aufruf 36
7.3 Abschnitte . 38
7.4 Feldmuster . . 39

II. WEITERE MÖGLICHKEITEN

8 Ketten und Koppeln

8.1 Verkettung . . . 47
8.2 Koppeln 48
8.3 Auswirkungen des Koppelns 50

9 Möglichkeiten zur Auswahl

9.1 Alternativen . . 53
9.2 Neue Bedingunen 55
9.3 Markierung . . 57

10 Behandlung von EUDAS-Dateien

10.1 Sortieren 59
10.2 Feldtypen 60
10.3 Reorganisieren 62

11 Variationen der Anzeige

11.1 Rollen 63
11.2 Feldauswahl 64
11.3 Übersicht . . . 64

12 Tragen

12.1 Einzelsatz 67
12.2 Tragen mit Prüfen . 68

13 Kopieren und Bearbeiten

13.1 Kopieren 73
13.2 Bearbeiten . . . 76
13.3 ELAN-Ausdrücke . . 77

14 Programmierung von Druckmustern

14.1 Abkürzungen 79
14.2 Komplexe Abkürzungen . . 84
14.3 Bedingte Musterteile . 87
14.4 Übersetzung 88

15 Drucken für Fortgeschrittene

15.1 Gruppen 91
15.2 Spaltendruck 96
15.3 Anschluß an die Textkosmetik . . . 97
15.4 Modi 98

III. DETAILBESCHREIBUNG

16 Menüs und Bedienung

16.1 Menüs . . 103
16.2 Auswahl 105
16.3 Hilfe und Dialog . 107
16.4 Menüstruktur 108
16.5 Dateioperationen . 109

17 Das virtuelle Dateimodell

17.1 Öffnen . . 113
17.2 Koppeln . 114
17.3 Änderungen 115
17.4 Sichern 117

18 Anzeigen und Ändern

18.1 Anzeigeformat . 119
18.2 Satzeditor . . 123
18.3 Suchmuster . 124

19 Drucken und Druckmuster

19.1 Druckmustersyntax 127
19.2 Der Druckvorgang 129
19.3 Interpretation von Musterzeilen 131
19.4 Anschluß zum ELAN-Compiler . . 134
19.5 Fehlermeldungen . 135

20 Verarbeitung

20.1 Feldstruktur ändern 137
20.2 Sortieren und Reorganisieren . 138
20.3 Kopieren . 139
20.4 Tragen 140
20.5 Nach Vorschrift ändern . 141

IV. EUDAS FÜR PROGRAMMIERER

21 Struktur von EUDAS-Dateien

21.1 Der Datentyp SATZ 145
21.2 Der Datentyp EUDAT 146
21.3 Satzposition . 147
21.4 Satzzugriffe 148
21.5 Sortieren und Reorganisieren . 149

22 Verwaltung der offenen Dateien

22.1 Dateiverwaltung 151
22.2 Feldstruktur . 154
22.3 Positionierung . 155
22.4 Änderungen 157
22.5 Suchbedingungen . . 158

23 Verarbeitungsfunktionen

23.1 Drucken . . 161
23.2 Kopieren . . 163
23.3 Tragen 164
23.4 Verarbeitung 167
23.5 Funktionen in Ausdrücken . . 168

24 Anzeige

24.1 Fensterverwalter 169
24.2 Anzeigegrundfunktionen 170
24.3 Editorfunktionen 171

25 Programmierung der Menüs

25.1 Menüformat 173
25.2 Verwaltung der Menüs . . 176
25.3 Aufruf . . . 177
25.4 Dialog . . . 179

ANHANG

A Fehlermeldungen 183
B Prozeduren mit Parametern . . . 187
C Register 191

I.

EINFÜHRUNG IN DIE BENUTZUNG

1

Was kann EUDAS ?

In diesem Kapitel wollen wir Ihnen erklären, was EUDAS Ihnen eigentlich bringen soll. Sie arbeiten sicher nicht nur aus Spaß am Computer, sondern wollen ihn für bestimmte Aufgaben einsetzen. Ein Computer kann bestimmte Aufgaben nur dann bearbeiten, wenn er dafür programmiert worden ist.

EUDAS ist nun ein Programm, das allgemein Aufgaben der Datenverwaltung lösen kann. Zunächst wollen wir Ihnen erläutern, wodurch dieses Anwendungsgebiet genau charakterisiert wird.

1.1 TEXTVERARBEITUNG UND DATENVERWALTUNG

Es gibt zwei Einsatzbereiche des Computers, die von fast jedem Anwender benötigt werden, egal auf welchem Spezialgebiet er tätig ist: nämlich die Textverarbeitung und die Datenverwaltung. Durch die Unterscheidung dieser beiden Bereiche werden die charakteristischen Merkmale der Datenverwaltung abgegrenzt.

Die Textverarbeitung befaßt sich damit, einen beliebigen Text auf einem Rechner zu erfassen und anschließend über einen Drucker wieder auszugeben. Der Sinn dieser Arbeit liegt darin, daß man einen einmal eingegebenen Text sehr einfach ändern kann. Außerdem kann der Computer einige unangenehme Aufgaben übernehmen, die beim Schreiben von Texten auftreten: die Aufteilung auf Zeilen und Seiten, die Seitennumerierung und vieles mehr.

Charakteristisch für die Textverarbeitung ist, daß der Einfluß des Computers sich auf kosmetische Details beschränkt. Die Speicherung und eventuelle Aufbereitung zum Drucken haben praktisch nichts mit dem Inhalt des Textes zu tun. Dies wäre für den Rechner auch sehr schwierig, da die im Text enthaltenen Informationen in menschlicher Sprache vorliegen, die für einen Rechner nicht verständlich ist.

Bei der Datenverwaltung werden ebenfalls textuelle Informationen gespeichert, diese liegen aber in einer aufbereiteten Form vor, die einen Teil des Inhalts für den Computer verständlich macht. Bei der Datenverwaltung werden Objekte betrachtet, die verschiedene Eigenschaften haben können. Ein solches Objekt kann z.B. eine Person sein. Als Eigenschaften werden mit dieser Person zusammenhängende Informationen betrachetet, die für die jeweilige Anwendung wichtig sind.

Als Beispiel für eine solche Betrachtungsweise ist der Arbeitnehmer im Betrieb. Einige der typischerweise erfaßten Daten sind Name, Adresse, Telefon, Geburtsdatum und Geschlecht. Alle diese Daten sind Eigenschaften oder **Attribute**, die einem bestimmten Menschen mehr oder weniger fest zugeordnet sind.

Die Betonung inhaltlicher Beziehungen erleichtert es dem Computer, die gespeicherten Daten in verschiedenen Variationen auszuwerten.

Um die Unterscheidung zwischen Textverarbeitung und Datenverwaltung deutlicher zu machen, werden im folgenden Beispiel die Informationen über eine Person in zwei unterschiedlichen Formen dargeboten, die für den Menschen die gleiche Aussagekraft haben:

1. Frau Magdalene Kant, geb. Hagedorn, wurde am 12. Januar 1946 geboren. Sie wohnt in Bonn in der Meckenheimer Allee 112. Seit 1977 arbeitet sie in unserer Firma. Sie ist telefonisch erreichbar unter der Nummer 0228/356782.

2.

Name:	Magdalene
Vorname:	Kant
Geburtsname:	Hagedorn
Geburtsdatum:	12.01.46
Geschlecht:	weiblich
Strasse:	Meckenheimer Allee 112
PLZ:	5200
Wohnort:	Bonn 1
Vorwahl:	0228
Telefon:	356782
beschäftigt seit:	1977

Die Form der Darstellung wie in der ersten Alternative eignet sich nur für den Menschen, da die gleiche Information auf viele verschiedene Weisen ausgedrückt werden könnte (z.B. unterschiedlicher Satzbau). Die zweite Alternative beschränkt

sich auf die für die bestimmte Anwendung wesentlichen Zusammenhänge; der Computer kann die Aufteilung der Information in einzelne Attribute ausnutzen.

In dieser zweiten Form können Sie Daten mit EUDAS erfassen und auch auswerten. Die Attribute können Sie jeweils passend zu den erfaßten Daten selbst bestimmen.

Es besteht nun die Möglichkeit, die erste Darstellung in unserem Beispiel aus der zweiten automatisch zu erzeugen, indem Sie dem Computer den Satzbau mit entsprechenden Platzhaltern vorgeben. Der Rechner setzt die einzelnen Attribute dann an die angegebenen Stellen. Diese Funktion ist ein Kernstück von EUDAS und wird in Abschnitt 1.3 näher erläutert.

1.2 EUDAS ALS KARTEIKASTEN

Wie Sie vielleicht schon bemerkt haben, ähnelt die zweite Form der Darstellung einer Karteikarte, auf der Platz für bestimmte Einträge freigehalten wird. Anhand dieses Modells können Sie sich in vielen Fällen die Arbeitsweise von EUDAS veranschaulichen. Sie sollten die Analogie allerdings nicht zu weit treiben: EUDAS schaufelt ja nicht wirklich mit Karteikarten herum. Manche Funktionen sind eben computerspezifisch und ließen sich mit Karteikarten gar nicht durchführen.

Mit EUDAS können Sie die möglichen Einträge auf den Karteikarten (also die Attribute) völlig frei bestimmen; die einzige Beschränkung besteht darin, daß Sie in einem Karteikasten nur Karten mit völlig gleichem Aufbau verwenden können. Wenn Sie eine neue Karteikarte entwerfen wollen, brauchen Sie nur Namen für die einzelnen Einträge anzugeben. EUDAS zeigt Ihnen dann quasi eine Karte am Bildschirm, in der diese Einträge aufgeführt sind.

Sie können nun am Bildschirm Daten auf diese Karteikarten schreiben. Dabei dürfen die Einträge fast beliebig lang sein; wenn der Platz auf dem Bildschirm nicht reicht, können Sie sich Ihre überdimensionale Karteikarte in Ausschnitten ansehen.

Die einmal eingegebenen Daten bleiben nun so lange gespeichert, wie Sie wollen (bzw. bis Ihr Rechner zusammenfällt). Haben Sie beim Eintragen Fehler gemacht, können Sie diese jederzeit korrigieren oder später noch weitere Informationen ergänzen.

```
          +-------+-----+
       +-------+-----+__!
    +-------+-----+__! __!
    ! Name   ! _____! __! __!
    ! Vorname! _____! __!.:'
  .'! Strasse! _____!.:    !
 +---------------+     .:'
 !  Kartei A     ! .:'
 +---------------+'
                          +-------+-----+
                        +-------+-----+__!
                     +-------+-----+__! __!
                     ! Name   ! _____! __! __!
                     ! Vorname! _____! __!.:'
                   .'! Strasse! _____!.:    !
                  +---------------+     .:'
                  !  Kartei B     ! .:'
                  +---------------+'
```

Abb 1-1: EUDAS als Karteikasten

Mit den gespeicherten Daten können Sie nun verschiedene Dinge anstellen (bzw. vom Rechner anstellen lassen). Das Einfachste ist natürlich das, was Sie mit einer Kartei auch machen würden, sich nämlich einzelne Karteikarten anzuschauen.

Um eine bestimmte Karteikarte herauszufinden, geben Sie EUDAS einfach den Inhalt vor, nach dem gesucht werden soll. Hier zeigt sich bereits der erste Vorteil eines Computers: Die Suche in der EUDAS-Kartei ist viel schneller, als Sie es von Hand könnten. Außerdem kann der Rechner keine Karte zufällig übersehen.

EUDAS zeigt sich auch dann überlegen, wenn Sie einen ganz bestimmten Teil der Kartei durchforsten müssen. Eine Bücherei muß z.B. regelmäßig alle Bücher heraussuchen, deren Leihfrist überschritten ist. Der Computer durchsucht in solchen Fällen ermüdungsfrei auch große Datenmengen.

Wenn Sie die Karteikarten in einer bestimmten Reihenfolge haben wollen, kann EUDAS auch das Sortieren übernehmen. Weitere automatische Vorgänge betreffen z.B. das Rauswerfen überflüssiger oder veralteter Karten. Die Einträge können auch nach einer bestimmten Vorschrift alle geändert werden. Solche Aufgaben treten z.B. in der Schule auf, wo die Schüler jedes Jahr versetzt werden müssen (natürlich bis auf Ausnahmen).

Auch Beziehungen zwischen verschiedenen Karteien kann EUDAS herstellen. Dies kann man noch einmal an dem Beispiel der Bücherei illustrieren. Wenn ein Buch gefunden wurde, dessen Leihfrist überschritten ist, muß der zugehörige Ausleiher gefunden und angeschrieben werden. Das Heraussuchen beider Karten kann EUDAS in einem Arbeitsgang durchführen.

1.3 DRUCKEN

Eine besondere Stärke von EUDAS ist die Möglichkeit, die gespeicherten Daten in schriftlicher Form auszuwerten. Dadurch, daß die Daten in einer Form gespeichert sind, die den Inhalt widerspiegelt, können die gleichen Daten in vielen verschiedenen Formen auf Papier ausgegeben werden.

Zu diesem Zweck geben Sie EUDAS ein Muster des gewünschten Ausdrucks vor. Der Rechner setzt dann an entsprechend markierten Leerstellen die gespeicherten Informationen ein und druckt das Ergebnis aus. Auf diese Weise ersparen Sie sich die umfangreiche Schreibarbeit, die anfällt, wenn die Informationen auf den Karteikarten in anderer Form benötigt werden.

Natürlich müssen Sie zum Entwerfen des Formulars kein gewiefter Programmierer sein. Wenn Sie einen Rundbrief verschicken wollen, schreiben Sie den Brief, als wollten Sie Ihn nur einmal schicken. Lediglich im Adressfeld müssen Sie Platzhalter an den Stellen vorsehen, an denen später die wirklichen Adressen stehen sollen.

```
+----------------------------------------+
!                                        !
!     XXXXXXXXXXXX                       !
!     XXXXXXXXXXXX                       !
!                                        !
!     XXXX XXXXXXXXXXXXX                 !
!                                        !
!     Lieber XXXXXXX !                   !
!                                        !
!     Dies ist ein Beispiel für ein      !
!     Druckmuster.                       !
!                       Viele Grüße      !
!                                        !
+----------------------------------------+
```

Abb. 1-2: Druckmuster

Die Möglichkeiten für solche Formulare sind unbegrenzt. Beispiele sind Briefe, Adreßaufkleber, Überweisungsaufträge und Kontrollisten. Mit den Inhalten einer Kartei können Sie beliebig viele verschiedene Ausgaben erzeugen. Bei dem obigen Beispiel der Leihbücherei könnten Sie EUDAS dazu einsetzen, nicht nur die säumigen Ausleiher herauszufinden, sondern die Mahnung gleich fertig für einen Fensterbriefumschlag herzustellen. Für den Bediener bliebe die einzige Tätigkeit, diesen Vorgang anzuwerfen.

Die Druckfunktion von EUDAS kann jedoch nicht nur zum Ausfüllen von Formularen verwendet werden. Wenn Sie Berechnungen anstellen oder Auswertungen vornehmen wollen, können Sie im Druckmuster auch ELAN-Anweisungen verwenden. Damit haben Sie eine komplette Programmiersprache für Ihre Muster zur Verfügung.

Ehe Sie einen Schreck bekommen: Selbst für komplizierte Muster brauchen Sie nur einen ganz kleinen Teil von ELAN zu beherrschen, da die meiste Arbeit immer von EUDAS übernommen wird (Sie müssen also nicht etwa selber ein ganzes Programm schreiben).

Anwendungen für diese Möglichkeit gibt es genug. Angefangen von einfachen Zählungen bis hin zu statistischen Auswertungen, von einfachen Summen bis zum kompletten Rechnungsschreiben. Immer nimmt Ihnen EUDAS alles das ab, was automatisch ablaufen kann. Sie versorgen EUDAS nur noch mit den passenden Formeln für Ihre Anwendung.

1.4 GRENZEN

Natürlich können Sie nicht alle Probleme mit EUDAS gleichermaßen gut lösen. EUDAS verwendet ein einfaches Modell (Karteikasten) und versucht, mit möglichst wenig Informationen von Ihrer Seite auszukommen. Kompliziertere Sachverhalte verlangen auch kompliziertere Strukturen, die Sie dann selbst entwerfen müssen. Eine einfache Lösung mit EUDAS kann in solchen Fällen zu langsam oder zu umständlich sein.

Wenn Sie jedoch die wenigen Strukturprinzipien von EUDAS verstanden haben, werden Sie sehr schnell viele Probleme mit EUDAS lösen können. Zuerst erfassen Sie einfach alle Daten, die Sie brauchen und überlegen sich erst dann, in welcher Form Sie diese Daten haben wollen. Auch nachträglich können Sie jederzeit noch neue Daten und Formulare hinzufügen, so daß Sie mit der Zeit EUDAS gewinnbringend für viele Routineaufgaben benutzen werden.

2
Installation

Bevor Sie EUDAS auf Ihrem System benutzen können, müssen Sie das Programm zuvor installieren. Wenn EUDAS schon auf Ihrem System zur Verfügung steht, können Sie dieses Kapitel getrost überlesen.

2.1 LIEFERUMFANG

EUDAS wird auf einer Diskette geliefert, die alle notwendigen Programme enthält. Um den Inhalt der Diskette feststellen zu können, starten Sie Ihr System und bringen es dazu, daß "gib kommando:" erscheint. Dann legen Sie die Diskette ein und geben das Kommando

```
archive ("EUDAS"); list (archive); release (archive)
```

Anschließend erscheint eine Übersicht der auf dem Archiv vorhandenen Programme. Folgende Programme sollten sich in dieser Übersicht wiederfinden:

```
"eudas.1"
"eudas.2"
"eudas.3"
"eudas.4"
"eudas.init"
"eudas.gen/s"
"eudas.gen/m"
"Adressen"
```

Eventuell können noch weitere Namen in der Übersicht auftauchen. Sollten einer der angegebenen Namen nicht vorhanden sein, reklamieren Sie die Diskette. Falls Sie statt der Übersicht eine Fehlermeldung erhalten, sollten Sie überprüfen, ob die Diskette das richtige Format besitzt oder Ihr Diskettenlaufwerk Probleme bereitet.

Wenn Sie so den Inhalt der Diskette kontrolliert haben, können Sie EUDAS installieren. Je nachdem, ob Sie ein Single-User oder ein Multi-User System benutzen, sind die Anweisungen unterschiedlich. Sie brauchen nur den Sie betreffenden der beiden folgenden Abschnitte zu lesen. Falls

Sie nicht wissen, welches System Sie benutzen: ein Multi-User System wird auf der Systemdiskette durch die Kennzeichnung "EUMEL 1.7.x/M" identifiziert, bei einem Single-User System steht als letztes Zeichen ein "S".

2.2 SINGLE-USER

Dieser Abschnitt betrifft Sie nur, wenn Sie EUDAS auf einem Single-User System installieren wollen.

Sie können EUDAS immer nur auf einer bestimmten Hintergrunddiskette installieren. Auf dieser Diskette sollten noch mindestens 300KB frei sein (stellen Sie dies durch das Kommando 'storage info' sicher). EUDAS kann anschließend auch nur auf dieser Diskette verwendet werden.

Starten Sie nun die gewünschte Diskette. Dann legen Sie die Diskette, auf der EUDAS sich befindet, in das Archivlaufwerk. Geben Sie dann das Kommando

```
archive("EUDAS"); fetch("eudas.gen/S", archive); run
```

Sie haben damit das Generatorprogramm gestartet, das die Installation automatisch durchführt. Lassen Sie während dieses Vorganges das EUDAS-Archiv eingelegt. Sie werden benachrichtigt, wenn die Generierung abgeschlossen ist.

Wenn Sie EUDAS auf allen Ihren Hintergrunddisketten haben möchten, können Sie das so erzeugte System als Muttersystem sichern. Mit dem Kommando 'save system' können Sie den Hintergrund komprimiert auf eine leere Archivdiskette schreiben. Mit dieser Sicherung können Sie dann jederzeit neue Systemdisketten wie von Ihrem Originalsystem herstellen.

2.3 MULTI-USER

Dieser Abschnitt betrifft Sie nur, wenn Sie EUDAS auf einem Multi-User System installieren wollen.

EUDAS muß in einer bestimmten Task installiert werden. Alle neuen Söhne und Enkel dieser Task können dann EUDAS aufrufen. Im Normalfall wird diese Task "PUBLIC" sein.

Zum Installieren müssen Sie in diese Task gehen (in diesem Beispiel "PUBLIC"). Dazu rufen Sie mit SV den Supervisor und geben das Kommando

```
continue ("PUBLIC")
```

Stelle Sie mit Hilfe des 'storage info'-Kommandos fest, ob auf Ihrem Hintergrund noch mindestens 300 KB frei sind (dieser Platz wird zur Generierung benötigt). Dann legen Sie die EUDAS-Archivdiskette ein und geben folgendes Kommando

```
archive("EUDAS"); fetch("eudas.gen/M", archive); run
```

Falls die Task "PUBLIC" Söhne besitzt, werden Sie gefragt, ob Sie diese löschen wollen. EUDAS steht nämlich nur in den Söhnen zur Verfügung, die nach der Installation eingerichtet wurden. Sie können die Generierung zu diesem Zeitpunkt auch noch abbrechen und zunächst die Söhne aufräumen.

Anschließend wird die automatische Generierung gestartet. Lassen Sie dabei das Archiv eingelegt. Die Generierung ist beendet, wenn das EUMEL-Bild erscheint. Die Task, in der die Generierung stattfindet, wird automatisch zu einer Managertask, das heißt, daß man von Ihr Söhne einrichten kann.

Sie können das so erzeugte System auch mit 'save system' auf einer Archivdiskette sichern. Lesen Sie dazu die Hinweise zur Systemsicherung im Systemhandbuch.

3
Bedienung der Menüs

Sie können EUDAS aufrufen, indem Sie bei "gib kommando:" das Kommando 'eudas' geben:

```
gib kommando:
eudas
```

Vergessen Sie nicht, die RETURN-Taste zu drücken. Nachdem Sie EUDAS aufgerufen haben, erscheint folgendes Menü:

```
:--------------------- ( EUDAS - Hauptmenue ) ---------------------------:
: a - aktuelle Datei ansehen        b - aktuelle Datei bearbeiten        :
: o - neue Datei oeffnen            f - Feldstruktur aendern             :
: k - Dateien koppeln               r - EUDAS-Datei sortieren            :
: s - aktuelle Dateien sichern      v - Dateiverwaltung                  :
:----------------------------------------------------------------------:
 CURSOR - Waehlen  LEER - Ausfuehren  ESC q - Verlassen  ESC ? - Hilfe
:----------------------------------------------------------------------:
```

Bevor Sie mit der tatsächlichen Arbeit anfangen, müssen Sie zuerst lernen, wie man solche Menüs bedient. Dies ist der Inhalt des folgenden Kapitels.

3.1 MENÜPRINZIPIEN

Ein Menü ist eine Auswahl für einige verschiedene Funktionen. Die Funktionen sind jeweils benannt und haben ein Minuszeichen davor. Im obigen Menü gibt es also 8 Funktionen zur Auswahl. Eine davon ist zum Beispiel

```
- neue Datei oeffnen
```

Eine der wichtigsten Regeln für die Bedienung von EUDAS ist, daß eine Zusammenfasssung der möglichen Tastendrücke in der **Statuszeile** in der Mitte des Bildschirms aufgeführt ist. Diese

Statuszeile befindet sich in dem Beispiel direkt unter dem Menürahmen. Wenn Sie also nicht wissen, was Sie für Tasten drücken sollen, schauen Sie immer in die Statuszeile!

Wir wollen die Informationen in der Statuszeile, die natürlich komprimiert sein müssen, an diesem Beispiel kennenlernen.

Die erste Information lautet:

```
CURSOR - Waehlen
```

Die Bezeichnung CURSOR ist eine Zusammenfassung für die vier Pfeiltasten, mit denen Sie den Cursor (die Schreibmarke auf dem Bildschirm) bewegen können. Sie können den Cursor allerdings nur auf den Minuszeichen bewegen. Probieren Sie dies am besten einfach aus.

Mit der Bewegung des Cursors lösen Sie noch keine Funktion aus. Erst wenn Sie die Leertaste drücken, wird die Funktion ausgeführt, vor der der Cursor gerade steht. Sie positionieren den Cursor also zuerst vor die gewünschte Funktion und drücken dann die Leertaste. Der zweite Teil aus der Statuszeile lautet daher

```
LEER - Ausfuehren
```

Um diese Funktion auszuprobieren, positionieren Sie bitte zum Testen auf die Funktion in der rechten unteren Ecke und drükken zur Ausführung die Leertaste. EUDAS reagiert darauf, indem er ein neues Menü ausgibt. Funktionen, die in Menüs aufgeführt sind, können also auch neue Menüs sein.

Um jetzt wieder in das alte Menü zurückzukehren, müssen Sie den dritten Teil der Statuszeile benutzen. Dort steht

```
ESC q - Zurueck
```

"ESC q" bedeutet, daß Sie zuerst die ESC-Taste und anschließend das kleine 'q' drücken müssen. Die ESC-Taste wird als Vorschalttaste verwendet, um die Funktion vom kleinen 'q' selbst zu unterscheiden. Sie werden die ESC-Taste in Zukunft noch häufiger antreffen.

Das 'q' steht für 'quit', also Beenden. ESC 'q' wird generell in EUDAS dazu benutzt, um in das vorige Menü zurückzukehren.

3.2 HILFESTELLUNG

Die letzte Information in der Statuszeile, nämlich

ESC ? - Hilfe

dient dazu, um Hilfsinformationen abzurufen. Denken Sie auch hier daran, die ESC-Taste und das Fragezeichen hintereinander zu drücken. Diese Tastenkombination kann ebenfalls in EUDAS jederzeit aufgerufen werden, um gefahrlos Informationen über die vorliegende Situation abzurufen.

Probieren Sie dies aus, indem Sie den Cursor vor irgendeine Funktion stellen. Sobald Sie ESC '?' gedrückt haben, erscheint anstatt des Menüs eine Information über diese Funktion. Auch die Statuszeile hat sich geändert.

Die Information zu der betreffenden Funktion kann nämlich auch aus mehreren Seiten bestehen. Es gibt zwei neue Tastenkombinationen, mit denen Sie die nächste oder die vorige Seite abrufen können.

Auf ESC 'w' (Weiter) erscheint die nächste Seite - natürlich nur, wenn vorhanden. Beachten Sie auch hier wieder die Tastenkombination mit ESC, die hintereinander gedrückt werden muß.

In ähnlicher Weise zeigt ESC 'z' (Zurück) wieder die vorige Seite. Mit der bekannten Tastenfolge ESC 'q' können Sie die Anzeige der Hilfsinformationen beenden. Es erscheint wieder das Menü wie vorher mit der alten Statuszeile.

Probieren Sie den Hilferuf auch bei den anderen Funktionen aus. Sie werden sehen, daß jedes Mal eine entsprechende Information erscheint.

3.3 STRUKTUR DER MENÜS

Wie Sie oben schon gesehen haben, kann eine Funktion in einem Menü zu einem neuen Menü führen. Aus diesem gelangt man dann durch ESC 'q' wieder in das alte Menü zurück. Natürlich kann sich das Spiel in dem Untermenü wiederholen: dort kann noch ein weiteres Menü aufgerufen werden.

Diesen Zusammenhang zwischen den Menüs kann man durch eine hierarchische Struktur veranschaulichen:

```
                    Hauptmenue
                     /       \
aktuelle Datei ansehen   aktuelle Datei bearbeiten
          |                        |
Einzelsatz bearbeiten    Dateiverwaltung
```

Die Grafik zeigt den Zusammenhang der fünf in EUDAS vorhandenen Menüs. Die darin vorkommenden Funktionen werden in diesem Handbuch noch besprochen. Wichtig für den Moment ist, daß die "Tiefe" eines Menüs in der Überschrift veranschaulicht wird.

Wie Sie oben gesehen haben, steht neben dem Menünamen jeweils eine spitze Klammer. Die Anzahl der Klammern geben die Tiefe an. Wenn Sie nochmal wie eben das Untermenü aufrufen, werden Sie sehen, daß dort an jeder Seite zwei Klammern stehen.

Falls Sie also nicht wissen, wo Sie sich in der Menüstruktur befinden, müssen Sie nur auf die Überschrift schauen. Dort sehen Sie, wie oft Sie ESC 'q' drücken müssen, um in das Hauptmenü zurückzukehren.

Auf diese Weise können Sie bei Problemen immer wieder zum Hauptmenü als rettenden Anker zurückkehren und von dort einen neuen Versuch starten.

3.4 ZUSAMMENFASSUNG

Zum Schluß wollen wir die wichtigsten Erkenntnisse aus diesem Kapitel zusammenfassen:

1. Menüs dienen zur Auswahl von Funktionen.
2. Die Statuszeile führt mögliche Tastendrücke auf.
3. Die Pfeiltasten bewegen nur den Cursor (Schreibmarke).
4. Die Leertaste führt eine Funktion aus.
5. ESC 'q' (Quit) führt eine Stufe zurück.
6. ESC '?' liefert spezifische Hilfestellung.
7. Die Menüs bilden eine hierarchische Struktur mit dem Hauptmenü an der Spitze.

4
EUDAS-Dateien

Bevor Sie EUDAS benutzen können, wollen wir Ihnen in diesem Kapitel die Terminologie von EUDAS vorstellen. Das Karteikartenmodell des ersten Kapitels muß ja auf einem Rechner realisiert werden. Dazu sollen zuerst einige Ausdrücke erklärt werden.

4.1 TEXTDATEIEN UND EUDAS-DATEIEN

Der wichtigste Ausdruck, der uns noch sehr häufig begegnen wird, ist Datei. Eine Datei ist eine Sammlung von Informationen in einem Computer, die als ein Objekt transportiert und behandelt werden können. Dieser Name wurde analog zu "Kartei" gebildet, mit dem Unterschied, daß eine "Daten-Kartei" gemeint ist.

Jede Datei hat einen eigenen Namen, durch den Sie identifiziert wird. Der Name ist meistens in Anführungsstriche eingeschlossen - die Anführungsstriche gehören jedoch nicht zum Namen. Der Name ist also wie die Aufschrift auf einem Aktenordner. Wenn alle Ordner im Schrank stehen, können Sie anhand des Namens den richtigen Ordner finden, anderenfalls müßten Sie alle Ordner öffnen.

Dateien werden nun für viele verschiedene Arten von Informationen benutzt. Sie können einerseits Texte enthalten oder auch Karteien, Grafiken, Formeln oder Zahlenkolonnen. Da wir bereits im ersten Kapitel den Unterschied zwischen Datenverwaltung und Textverarbeitung besprochen haben, sind in diesem Zusammenhang die beiden ersten Verwendungsweisen wichtig.

Textdateien sind Dateien, die normale Texte enthalten, die mit dem Editor verändert und einfach ausgedruckt werden können. In Ihnen werden also Informationen gespeichert, wie die Textverarbeitung sie benötigt.

EUDAS-Dateien sind dagegen Dateien, die Informationen in der Form von Karteikarten enthalten. Sie haben also die Struktur, wie sie von der Datenverwaltung benötigt wird.

Es ist sehr wichtig, diese beiden Typen von Dateien auseinanderzuhalten. Der Computer kann nämlich teilweise beide Arten von Dateien gleichbehandeln, und zwar dann, wenn der Inhalt der Dateien nicht betrachtet werden muß.

Ein häufiger Fall ist zum Beispiel, wenn Dateien zur Sicherung auf eine Diskette geschrieben werden sollen. In diesem Fall genügt die Angabe des Namens, dem Rechner ist es egal, welchen Inhalt die Datei hat.

Anders ist es aber, wenn Sie den Inhalt betrachten wollen. Dazu brauchen Sie dann ein Programm, das die innere Struktur der Datei kennt. Textdateien können Sie zum Beispiel mit dem Editor ansehen. EUDAS-Dateien müssen Sie jedoch mit EUDAS ansehen, da der Editor die EUDAS-Struktur nicht kennt. Es ist also in vielen Fällen sinnvoll, durch den Namen der Datei zu kennzeichnen, ob es sich um eine Textdatei oder eine EUDAS-Datei handelt.

4.2 AUFBAU VON EUDAS-DATEIEN

EUDAS verwendet bestimmte Ausdrücke, um die Strukturelemente einer EUDAS-Datei zu kennzeichnen. Diese Namen wurden nun nicht direkt aus dem Karteikartenmodell übernommen, da es auch noch andere Modelle gibt und keine falschen Assoziationen auftreten sollen.

EUDAS verwendet die Bezeichnung **Satz** für eine Karteikarte. Eine EUDAS-Datei besteht also aus einer Anzahl von gleichartigen Sätzen.

Jeder Satz ist unterteilt in sogenannte **Felder**. Ein Feld entspricht einem Attribut bzw. einem Eintrag auf der Karteikarte. Ein Feld ist wiederum unterteilt in einen **Feldnamen** und einen **Feldinhalt**.

Der Feldname identifiziert ein bestimmtes Feld innerhalb eines Satzes. Die Feldnamen sind natürlich für alle Sätze gleich. Die Feldnamen einer EUDAS-Datei sind beliebig und können von Ihnen selbst festgelegt werden.

Der Feldinhalt enthält die eigentliche Information des entsprechenden Attributs. Der Feldinhalt darf ebenfalls aus beliebig vielen Zeichen bestehen.

```
        +------+------+------+------
        !Satz 1!Satz 2!Satz 3!Satz 4  ...
+------+------+------+------+------
!Feld A!      !      !      !
!Feld B!      !      !      !
!Feld C!      !      !      !        ...
!Feld D!      !      !      !
!Feld E!      !      !      !
+------+------+------+------+------
```

Abb. 4-1 Struktur einer EUDAS-Datei

Aus technischen Gründen gibt es natürlich auch einige Beschränkungen, die hier nicht verschwiegen werden sollen. Eine Datei kann maximal etwa 5000 Sätze enthalten, ein Satz darf aus maximal 256 Feldern bestehen. Die einzelnen Sätze in der EUDAS-Datei werden durch ihre jeweilige Positionsnummer identifiziert, also quasi von 1 bis 5000 durchnumeriert.

4.3 ARCHIV

Bevor wir uns eine EUDAS-Datei in der Realität ansehen können, müssen wir uns erst eine besorgen. Normalerweise werden Sie sich später eigene Dateien erstellen. Da dies aber im Moment noch zu schwierig ist, müssen wir auf eine fertige Datei zurückgreifen.

Eine solche Datei befindet sich auf dem EUDAS-Archiv. Sie trägt den Namen "Adressen". Diese Datei sollen Sie nun von der Archivdiskette ins System holen. Dabei werden Sie gleichzeitig noch eine ganze Menge über die Bedienung von EUDAS lernen.

Dazu gehen Sie vom EUDAS-Hauptmenü in das Menü "Dateiverwaltung". Dort sollten Sie beim Ausprobieren im letzten Kapitel schon kurz gewesen sein. Dieses Menü besteht aus zwei Hälften: die linke Hälfte betrifft Dateien auf einer Archivdiskette - die rechte Hälfte bezieht sich auf Dateien auf dem System.

Legen Sie nun das EUDAS-Archiv ein und wählen Sie die Funktion

- Archivuebersicht

Sie bekommen nach kurzer Wartezeit alle auf dem Archiv vorhandenen Dateien angezeigt. Wenn Sie das richtige Archiv eingelegt haben, sollte dort auch der Name "Adressen" auftauchen.

Nachdem Sie die Übersicht nit ESC 'q' verlassen haben, wählen Sie als nächstes die Funktion

- Datei vom Archiv holen

Sie werden nun nach dem Namen der gewünschten Datei gefragt. Sie könnten jetzt den Namen direkt eingeben.

Um Ihnen eine neue Möglichkeit kennenzulernen, die Sie sicher oft verwenden werden, drücken Sie bitte ESC 'z'. Diese Tastenkombination, die auch in der Statuszeile angegeben ist, zeigt Ihnen eine Auswahl aller vorhandenen Dateien.

Die Auswahl sollte nach einem kurzen Moment erscheinen. Der untere Teil des Bildschirms sieht dann etwa so aus:

```
:-----------------------------------------------------------------------:
 'x' - Ankreuzen  'o' - Durchstreichen  ESC q - Verlassen  ESC ? - Hilfe
:-----------------------------------------------------------------------:
 Auswahl der Dateien auf dem Archiv. Gewuenschte Datei(en) bitte ankreuzen:
:-----------------------------------------------------------------------:
      o "eudas.1"
      o "eudas.2"
      o "eudas.3"
      o "eudas.4"
      o "eudas.init"
      o "eudas.gen/s"
      o "eudas.gen/m"
      o "Adressen"
:-----------------------------------------------------------------------:
```

Auf den Kreisen können Sie den Cursor auf und ab bewegen. Mit 'x' können Sie dann die gewünschte Datei ankreuzen, bei Bedarf auch mehrere. Probieren Sie dies ruhig aus. Wenn Sie eine Datei angekreuzt haben, können Sie mit 'o' die Ankreuzung wieder löschen.

Sie sehen, wie gleichzeitig mit dem Ankreuzen Nummern neben den Namen erscheinen. Wenn Sie Ankreuzungen wieder löschen, werden die Nummern entsprechend korrigiert. Die Nummern geben an, in welcher Reihenfolge die Dateien nachher genommen werden.

Löschen Sie jetzt alle Auswahlen, bis nur noch "Adressen" angekreuzt ist (diese Datei wollten wir ja holen). Beenden Sie die Auswahl dann mit ESC 'q'. Die Datei wird anschließend vom

Archiv zum System kopiert und steht dort zur Verfügung. Daß die Datei angekommen ist, können Sie feststellen, indem Sie

- Dateiuebersicht

auswählen. Der Name taucht jetzt in dieser Übersicht auf. Wenn Sie eine Datei auf dem System nicht mehr benötigen, können Sie diese mit der Funktion

- Datei loeschen

entfernen. EUDAS fragt aber sicherheitshalber noch einmal nach, ehe er die Datei löscht.

Nach diesem kleinen Exkurs können Sie die Dateiverwaltung wieder verlassen, um nun endlich unsere EUDAS-Datei anzuschauen.

4.4 ÖFFNEN

Noch eine letzte Bitte um Geduld: Bevor Sie sich den Inhalt einer EUDAS-Datei anschauen können, müssen Sie die Datei wie einen Karteikasten öffnen. Nach dem Öffnen beziehen sich alle weiteren Funktionen auf die gerade geöffnete Datei. Aus diesem Grund darf auch immer nur eine einzige Datei geöffnet sein - als hätte auf Ihrem Schreibtisch nur ein Karteikasten Platz.

Das Öffnen ist eine der im Hauptmenü angegebenen Funktion. Positionieren Sie im Hauptmenü den Cursor auf die zweite Funktion in der ersten Spalte:

- neue Datei oeffnen

Dann drücken Sie die Leertaste, um die Funktion auszuführen. Als erstes erscheint im unteren Teil des Bildschirms eine Frage:

```
Wollen Sie die Datei nur anschauen (j/n) ?
```

Der Cursor bleibt hinter der Frage stehen, EUDAS wartet also auf eine Antwort von Ihnen. Beachten Sie auch die Änderung in der Statuszeile: Dort ist angegeben, wie Sie auf die Frage reagieren können. Die beiden Tastenkombinationen ESC 'q' und ESC '?' funktionieren auch hier.

In unserem Fall wollen wir an der Spieldatei nichts verändern, wir beantworten die Frage also mit einem 'j' (es ist nur

ein Tastendruck nötig). Als nächstes werden Sie nach dem Namen gefragt (Beachten Sie auch hier die Statuszeile).

Tippen Sie nun "Adressen", aber ohne die Anführungsstriche, ein. Beenden Sie die Eingabe mit RETURN. EUDAS öffnet die Datei und kehrt zum Menü zurück. Alternativ können Sie die Datei auch in einer Auswahl ankreuzen, wenn Sie ESC 'z' drücken. Wie Sie nun den Inhalt ansehen können, wird im nächsten Kapitel erklärt.

In den bisherigen Dialogen haben Sie zwei wichtige Formen der Anfrage erlebt. Die erste Anfrage erwartet ein Ja oder Nein ('j' oder 'n'), die zweite Frage wollte eine bestimmte Information anfordern. Beide Typen werden in anderen Funktionen immer wieder auftauchen. Daher ist es nützlich, sich mit ihnen vertraut zu machen.

Wenn Sie wollen, können Sie das Öffnen noch mehrmals aufrufen und dabei unterschiedliche Variationen der beiden Anfragen durchspielen. Sie sollten zum einen die Wirkungsweise von ESC 'q' und ESC '?' erproben, und zum anderen verschiedene Möglichkeiten bei der Namenseingabe ausprobieren.

Bei der Eingabe des Dateinamens können Sie nämlich bisher getipptes noch ändern, bevor Sie die RETURN-Taste drükken. Sollten Sie zum Beispiel ein falsches Zeichen eingegeben haben, können Sie mit den Cursortasten nach links und rechts gehen und das falsche Zeichen überschreiben.

Sie können auch die RUBOUT-Taste benutzen, um ein Zeichen zu löschen. Wollen Sie Zeichen einfügen, drücken Sie RUBIN. Anschließend befinden Sie sich im Einfügemodus, das heißt alle Ihre Eingaben werden automatisch eingefügt und überschreiben nicht.

Die Möglichkeiten bei der Eingabe von Namen entsprechen den Möglichkeiten des Editors; schauen Sie sich gegebenenfalls die Beschreibung des Editors an. Die Hilfe zur Eingabe liefert eine kurze Zusammenfassung der Möglichkeiten.

5
Ansehen und Suchen

In diesem Kapitel sollen Sie nun endlich etwas zu sehen bekommen: unsere EUDAS-Datei ist geöffnet und kann jetzt besichtigt werden.

5.1 BILDSCHIRMAUFBAU

Nach dem Öffnen steht der Cursor genau auf der Auswahl

- aktuelle Datei ansehen

Sie brauchen also nur die Leertaste zu betätigen, um diese Funktion auszulösen.

Im unteren Teil des Bildschirm erscheint zunächst eine Ausgabe mit einigen markierten Teilen. Anschließend wird im oberen Teil ein neues Menü ausgegeben, so daß sich etwa folgendes Bild ergibt:

```
:------------------------- << Ansehen >> --------------------------------:
: w - Satz weiter                  z - Satz zurueck                      :
: s - Selektion einstellen         l - Selektion loeschen                :
: d - direkt auf Satz              v - Bild vertikal verschieben         :
: e - Einzelsatz bearbeiten        b - aktuelle Datei bearbeiten         :
:-----------------------------------------------------------------------:
 CURSOR - Waehlen  LEER - Ausfuehren  ESC q - Verlassen  ESC ? - Hilfe
:-----------------------------------------------------------------------:
 Satz 1   ..................................................... Feld  1
 Name     Wegner
 Vorname  Herbert
 PLZ      5000
 Ort      Koeln
 Strasse  Kraemergasse 12
 m/w      m
 .......................................................................
```

Der untere Teil des Bildschirms zeigt jeweils einen Ausschnitt aus der Datei, und zwar genau einen Satz. Dieses Bild hat eine Überschrift, in der unter anderem die Satznummer angezeigt

wird. In unserem Beispiel sehen wir also den Satz 1 aus der Datei "Adressen". Die übrigen Teile der Überschrift brauchen uns im Moment noch nicht zu interessieren.

Unterhalb der Überschrift befindet sich der eigentliche Inhalt, und zwar sind links markiert die Feldnamen zu sehen, während rechts die zugehörigen Inhalte des betreffenden Satzes stehen. Dieses Bild ähnelt einer Karteikarte mit einem festen Format.

Die Abschlußzeile am Ende gibt an, daß für diesen Satz keine weiteren Informationen mehr vorhanden sind. Wir werden aber später noch sehen, wie man anderenfalls die unsichtbaren Informationen sichtbar machen kann.

5.2 BEWEGEN

Nun wollen Sie nicht immer nur einen Satz betrachten (das wäre ja furchtbar langweilig). Daher müssen Sie die Möglichkeit haben, sich in der Datei zu "bewegen". Dies geschieht mit Hilfe der beiden Funktionen

w - Satz weiter

und

z - Satz zurueck

Sie haben die Wirkung, daß der Satz mit der nächsthöheren bzw. nächstniedrigeren Satznummer angezeigt wird. Natürlich funktioniert dies nur, wenn noch ein Satz vorhanden ist: am Anfang (Satz 1) können Sie nicht zurückgehen. In diesem Fall ignoriert EUDAS Ihren Befehl einfach.

Wenn Sie bis zum Ende der Datei gehen (keine Angst - diese Datei enthält nur 10 Sätze), werden Sie feststellen, daß zum Schluß ein ganz leerer Satz erscheint. Dieser Satz ist eine Art Endemarkierung, er informiert Sie, daß keine weiteren Sätze vorhanden sind. Dieser Satz ist aber kein richtiger Satz, daher wird bei ihm als Satznummer "ENDE" ausgegeben. (Wenn Ihnen diese Art Endemarkierung merkwürdig erscheint: sie hat schon einen triftigen Grund).

Um einen Satz mit bekannter Satznummer gezielt anzuwählen, können Sie die Funktion

- direkt auf Satz

verwenden. Sie müssen anschließend die Satznummer eingeben (Bitte mit RETURN beenden). Ist der Satz vorhanden, erscheint dieser, ansonsten stehen Sie am Ende der Datei.

Vielleicht ist Ihnen inzwischen schon aufgefallen, daß vor jeder Funktion in einem Menü ein Buchstabe steht. Damit hat es folgendes auf sich: da das Positionieren des Cursors zum Auswählen einer Funktion mehrere Tastendrücke erfordern kann, haben Sie die Möglichkeit, jede Funktion auch über einen Buchstaben auszuführen.

Dies ist besonders dann sinnvoll, wenn Sie mit den eben besprochenen Funktionen schnell in der Datei "blättern" wollen. Andererseits müssen Sie sich aber für eine schnelle Reaktion auch einige der Tasten merken. Für die Praxis empfiehlt sich folgender Kompromiß: die meistgebrauchten Funktionen über Tasten und der Rest ganz normal.

5.3 SUCHEN

Stellen Sie sich vor, die Datei wäre größer und Sie müßten eine bestimmte Adresse heraussuchen. Dazu würden Sie durch die ganze Datei durchgehen, bis die gewünschte Adresse erscheint. Das wäre natürlich bei vielen Adressen eine ungeheuer mühselige Arbeit, die mit einem Karteikasten wahrscheinlich schneller zu erledigen wäre.

EUDAS bietet Ihnen jedoch die Möglichkeit, nach bestimmten Sätzen zu suchen. Dazu müssen Sie angeben, wonach gesucht werden soll. Als Beispiel wollen wir die Adresse von Frau Simmern suchen. Bewegen Sie sich zunächst zurück bis auf den ersten Satz. Dann wählen Sie die Funktion

- Selektion einstellen

Auf dem Bildschirm verschwinden die Feldinhalte und der Cursor steht hinter dem ersten Feldnamen. Dies bedeutet, daß Sie neben die Feldnamen etwas schreiben können. Auch in der Überschrift hat sich etwas geändert: statt "Feld 1" steht dort "zeile 1".

Als Angabe, was gesucht werden soll, schreiben Sie jetzt in der ersten Zeile neben "Name" die Bedingung "Simmern" (ohne Anführungsstriche). Sie haben jetzt ein einfaches Suchmuster angegeben. Ein Suchmuster besteht aus Bedingungen, die neben die Feldnamen geschrieben werden. Unser einfaches Suchmuster lautet übersetzt:

> Wähle alle Sätze aus, bei denen "Simmern" im Feld "Name" steht.

```
ESC RUBOUT - Zeile loeschen ESC a - Abbruch ESC q - Zurueck ESC ? - Hilfe
:-------------------------------------------------------------------------:
Satz 1    ................ "Suchmuster eingeben" ............... zeile  1
Name     Simmern
Vorname
PLZ
Ort
Strasse
m/w
.........................................................................
```

Abb. 5-1 Eingabe eines Suchmusters

Beenden Sie die Eingabe des Suchmusters mit ESC 'q'. Es erscheint wieder das vorherige Bild, mit dem Unterschied, daß jetzt in der Überschrift ein "SUCH-" auftaucht. EUDAS steht also immer noch auf dem ersten Satz und hat noch nicht gesucht.

Das Suchen beginnt erst, wenn Sie sich in der Datei bewegen. In diesem Fall erhalten die Funktionen "Satz weiter" und "Satz zurück" eine etwas geänderte Bedeutung. Sie gehen nämlich nicht einfach zum nächsten bzw. vorigen Satz, sondern zum nächsten bzw. vorigen **ausgewählten** Satz.

Als Indikator, daß Sie sich auf dem gesuchten Satz befinden, dient die Anzeige "SUCH+". Probieren Sie dies jetzt aus, indem Sie weitergehen. Als nächster Satz erscheint der gewünschte Satz 7. Die nicht ausgewählten Sätze wurden also übersprungen. Das gleiche passiert, wenn Sie noch weiter gehen. Da kein passender Satz mehr vorhanden ist, erscheint das Dateiende und die Anzeige "SUCH-".

Beachten Sie, daß das Einstellen der Suchbedingung nicht automatisch eine Suche bewirkt. Sie müssen die Suche selbst auslösen.

Sollten Sie sich nicht am Anfang der Datei befinden, wenn Sie eine Suchbedingung einstellen, kann es sein, daß sich der gesuchte Satz sowohl vor als auch hinter der aktuellen Position befindet. In diesem Fall müssen Sie entscheiden, ob Sie vorwärts oder rückwärts gehen. Wenn Sie rückwärts gehen und der Satz ist nicht in diesem Abschnitt, erscheint der erste Satz mit der Anzeige "SUCH-". Gehen Sie dann wieder vorwärts, finden Sie den Satz auf jeden Fall.

Die Funktion "direkt auf Satz" richtet sich natürlich nicht nach der eingestellten Bedingung, da Sie ja eine be-

stimmte Satznummer wünschen. Aus der "SUCH"-Anzeige können Sie jedoch entnehmen, ob die Suchbedingung auf diesen Satz zutrifft.

Um den Charakter einer Selektion nochmal deutlich zu machen, wollen wir jetzt eine Bedingung einstellen, die auf mehrere Sätze zutrifft. Dies hätte uns auch eben passieren können, wenn es mehrere Simmern gegeben hätte. Wir können zum Beispiel alle weiblichen Personen auswählen.

Dazu wählen Sie erneut die Funktion "Selektion einstellen". Die vorher eingestellte Bedingung wird in diesem Fall automatisch gelöscht. Schreiben Sie in das Feld "m/w" die Bedingung "w" (ohne Anführungsstriche) und verlassen Sie die Eingabe mit ESC 'q'.

Wenn Sie sich jetzt in der Datei bewegen, sehen Sie, daß immer nur weibliche Personen angezeigt werden - die männlichen werden unterdrückt (in Umkehrung der Realität).

```
        !  1  !     !  3  !  4  !     !  6  !
+-----+-----+-----+-----+-----+-----+-----+
!Name ! ..  !     ! ..  ! ..  !     !     !
!     !     !     !     !     !     !     !
   :     :           :     :           :
!     !     !     !     !     !     !     !
!m/w  ! w   !     ! w   ! w   !     ! w   !
+-----+-----+     +-----+-----+     +-----+
```

Abb. 5-2 Wirkung einer Selektion

Wollen Sie wieder alle Sätze sehen, müssen Sie die Funktion

- Selektion loeschen

auswählen. Die Anzeige "SUCH" verschwindet wieder, um anzudeuten, daß keine Suchbedingung mehr eingestellt ist.

5.4 SUCHBEDINGUNGEN

Im letzten Abschnitt haben Sie gesehen, wie das Einstellen einer Suchbedingung funktioniert. In diesem Abschnitt wollen wir weitere Möglichkeiten zur Formulierung von Suchmustern kennenlernen.

Die erste Möglichkeit kennen Sie schon. Wenn neben einen Feldnamen ein Text geschrieben wird, bedeutet dies, daß ausgewählte Sätze im Inhalt dieses Feldes mit dem Text übereinstimmen müssen.

Nun kann es sein, daß mehrere Bedingungen gelten müssen. Im ersten Beispiel des vorigen Abschnitts hätten wir zum Beispiel auch noch den Vornamen "Anna-Maria" angeben können, um bei mehreren Simmern die richtige auszuwählen. Wird also in mehrere Felder eine Bedingung geschrieben, müssen alle diese Bedingungen gleichzeitig zutreffen.

Würden Sie in unserem Beispiel noch als dritte Bedingung "m" für das Feld "m/w" angeben, würde gar kein Satz mehr ausgewählt, da Anna-Maria Simmern natürlich nicht männlich ist. Auch das kann also passieren.

```
Name      Simmern
Vorname   Anna-Maria
..
..
m/w       m
```

Abb. 5-3 Kombination von Bedingungen

Die Bedingungen, die wir bis jetzt kennengelernt haben, müssen alle ganz exakt zutreffen. Häufig tritt aber der Fall auf, daß der gesuchte Name nicht genau bekannt ist. In diesem Fall kann der Name im Suchmuster auch teilweise eingegeben werden. Der unbekannte Teil am Anfang oder am Ende wird einfach durch einen Stern markiert.

Wenn Sie also als Bedingung "Sim*" für den Namen angeben, so würde dies auf den Namen Simmern zutreffen, aber zum Beispiel auch auf Simmerath oder Simon. Die Bedingung "*mern" würde nicht nur auf Simmern zutreffen, sondern auch auf Pommern.

Der Stern kann aber auch für einen leeren Text stehen. So trifft "Simmern*" auf Simmern zu, aber auch auf Doppelnamen. die mit Simmern beginnen. Wissen Sie jetzt nicht, ob Simmern in dem Doppelnamen vorne oder hinten erscheint, können Sie auch an beiden Seiten einen Stern machen. Die Bedingung "*Simmern*" trifft also sowohl auf Deckerath-Simmern als auch auf Simmern-Jakob zu.

Es gibt noch eine Reihe von weiteren Möglichkeiten, Bedingungen im Suchmuster zu formulieren. Auch komplexe Kombinationen von Bedingungen sind möglich. Mit dem bisher Besprochenen sollten Sie aber in vielen Fällen auskommen. Die übrigen Möglichkeiten werden in einem späteren Kapitel erklärt. Schauen Sie bei Bedarf dort nach.

6
Einrichten und Ändern

6.1 NEUE DATEI EINRICHTEN

Nachdem Sie sich bis jetzt an einer existierenden Datei erprobt haben, können wir nun dazu übergehen, eine eigene neue Datei einzurichten. Als Beispiel wollen wir ein kleines Telefonregister erstellen.

Dazu gehen Sie wieder in das Hauptmenü zurück und wählen die Funktion "neue Datei oeffnen". Verneinen Sie die Frage, ob Sie die Datei nur ansehen wollen. Als Dateinamen geben Sie "Telefonnummern" an (denken Sie auch hier daran, daß die Anführungsstriche nur zur Abgrenzung dienen und nicht mit eingegeben werden sollen).

Da die Datei "Telefonnummern" noch nicht existiert, werden Sie gefragt, ob Sie diese Datei neu einrichten wollen. Es kann ja auch sein, daß Sie sich vertippt haben und eine andere, existierende Datei meinten. In unserem Fall wird die Datei aber wirklich neu eingerichtet, daher bejahen Sie die Frage.

Wenn Sie beim Öffnen eine Datei neu einrichten, müssen Sie zuerst die Feldnamen festlegen. Daher wird Ihnen jetzt Gelegenheit gegeben, die Feldnamen untereinander einzugeben. Zur Korrektur können Sie die gleichen Tasten verwenden wie im Editor.

Für unser Beispiel wären die folgenden Felder sinnvoll:

```
"Name"
"Vorname"
"Strasse"
"PLZ"
"Ort"
"Telefon"
"Bemerkungen"
```

Geben Sie die Namen in dieser Reihenfolge (ohne Anführungsstriche) ein. Beenden Sie die Eingabe mit ESC 'q'.

Damit Sie gegebenenfalls noch an den Feldnamen etwas ändern können (und für andere Zwecke, die hier nicht interessieren), wird Ihnen anschließend nochmal eine Auswahl aller

Feldnamen angeboten. Beenden Sie auch diese Funktion mit ESC 'q' (wir werden später noch sehen, was es damit auf sich hat). Sie werden danach gefragt, ob Sie die Prüfbedingungen ändern wollen. Da auch das erst später erklärt wird, geben Sie hier ein 'n' ein. Ihre Datei ist nun eingerichtet.

Sie haben gesehen, daß das Einrichten einer Datei eigentlich nur aus der Eingabe der Feldnamen besteht. Die anderen Funktionen werden Sie nur selten verwenden. Wenn Sie später noch Felder anfügen wollen, ist dies ohne weiteres möglich.

6.2 ÄNDERN UND EINFÜGEN

Nachdem die Datei nun eingerichtet worden ist, wollen wir einige Sätze eingeben. Wenn Sie das Menü "aktuelle Datei anzeigen" anwählen, sehen Sie nur den leeren Satz, der das Dateiende markiert. Um neue Sätze aufzunehmen, gibt es im Menü "Einzelsatz bearbeiten" die Funktion

- neuen Satz einfuegen

Wenn Sie diese Funktion aufrufen, geschieht etwas Ähnliches wie beim Eingeben des Suchmusters. Der Cursor wandert wieder hinter den ersten Feldnamen und in der markierten Überschrift erscheint die Aufforderung

"Satz einfuegen"

Sie können nun die Feldinhalte der einzelnen Felder neben die entsprechenden Feldnamen schreiben.

Um eventuelle falsche Eingaben zu korrigieren, können Sie die ähnlich wie im Editor mit den Cursortasten herumfahren und falsche Eingaben überschreiben. Sie beenden die Eingabe mit ESC 'q'. Anschließend ist der neue Satz vorhanden.

Die Funktion "Satz einfuegen" fügt den neuen Satz vor dem bisher angezeigten Satz ein. Dies erklärt auch, warum das Ende der Datei ein leerer Pseudosatz ist. Um nämlich am Ende der Datei einen neuen Satz anzufügen, muß man vor dem Endesatz einen Satz einfügen.

Nachdem Sie also jetzt den ersten Satz eingegeben haben, müssen Sie sich wieder zum Ende bewegen, damit der nächste Satz hinter dem ersten steht. Damit Sie deswegen nicht extra in ein anderes Menü gehen müssen, gibt es hierfür eine Abkürzung: Wenn Sie die Eingabe mit ESC 'w' (Weiter) statt ESC 'q'

beenden, geht EUDAS nach dem Einfügen des Satzes weiter zum nächsten und fügt dort wieder einen Satz ein.

Auf diese Weise können Sie also schnell eine ganze Reihe von Sätzen nacheinander eingeben. Nachdem Sie einen Satz eingegeben haben, drücken Sie ESC 'w' und können gleich anschließend schon mit der Eingabe des nächsten Satzes beginnen. Alle so eingegebenen Sätze erscheinen nachher in der Reihenfolge der Eingabe.

Wenn Sie nachträglich noch eingegebene Daten ändern wollen, können Sie die Funktion

- Satz aendern

verwenden. Sie haben anschließend wie beim Einfügen Gelegenheit, neue Daten einzugeben. Allerdings werden Ihnen die bisherigen Daten gleich mit angeboten, so daß Sie nur die Änderungen eingeben müssen. Alles andere kann unverändert bleiben. Auch diese Funktion wird mit ESC 'q' verlassen.

Probieren Sie die beiden Funktionen jetzt aus. Geben Sie zuerst einige Sätze ein (denken Sie an ESC 'w'). Anschließend können Sie einige dieser Sätze noch modifizieren. Versuchen Sie auch einmal, mitten in der Datei einen neuen Satz einzufügen.

Stellen Sie beim Ändern oder Einfügen fest, daß Sie irgendeinen krassen Fehler gemacht haben, können Sie die Operation mit ESC 'a' abbrechen. Beim Ändern bleibt dann der alte Zustand unverändert, beim Einfügen wird kein Satz eingefügt.

6.3 SATZEDITOR

In diesem Abschnitt sollen die Möglichkeiten bei der Eingabe von Satzinhalten genauer behandelt werden. Die gleichen Möglichkeiten stehen auch bei der Eingabe von Suchmustern zur Verfügung, die wir ja im vorigen Kapitel besprochen hatten. Der Programmteil, der dies ermöglicht, heißt Satzeditor.

Die Bedienung des Satzeditors ähnelt sehr stark der des EUMEL-Editors. Der beschreibbare Bereich ist jedoch kleiner. Er umfaßt das ganze Gebiet, in dem sonst die Feldinhalte erscheinen.

Wie beim Editor können Sie den Cursor mit den Cursortasten (Pfeiltasten) bewegen. Mit der Taste RUBOUT können Sie

ein Zeichen löschen. Die restlichen Zeichen der Zeile rücken dann nach. Mit RUBIN dagegen schalten Sie in einen Einfügemodus um. Alle eingegebenen Zeichen werden dann eingefügt - der Rest der Zeile rückt entsprechend nach rechts. Nochmaliges Drücken von RUBIN schaltet wieder in den alten Modus. Welcher Modus eingeschaltet ist, steht in der Überschriftzeile.

Auch den Lernmodus des Editors können Sie verwenden. Der Lernmodus wird beim Verlassen des Satzeditors jedoch nicht ausgeschaltet, Sie können aber dann im Menü und bei allen Eingaben weiterlernen. Informieren Sie sich gegebenenfalls über den Lernmodus im EUMEL-Benutzerhandbuch.

Mit der Kombination ESC RUBOUT können Sie den Rest einer Zeile ab der Cursorposition löschen. Steht der Cursor in der ersten Spalte, wird dementsprechend die ganze Zeile gelöscht.

Entsprechend fügen Sie mit der Funktion ESC RUBIN eine neue Zeile ein. Dies ist immer dann erforderlich, wenn ein Feldinhalt nicht auf eine Zeile paßt. Der Teil der Zeile, der hinter dem Cursor steht, wird bei ESC RUBIN in die neue Zeile mitgenommen.

Ein Hinweis für alle, die sich mit der Editorbedienung schon auskennen: EUDAS benutzt den Editor als Unterprogramm. Jedoch haben einige Editorfunktionen unliebsame Auswirkungen. Besonders gefährlich sind hier HOP RUBOUT und HOP RUBIN. Diese Funktion zerstören die Korrespondenz zwischen Feldnamen und Feldinhalten, das heißt der Feldinhalt steht nicht mehr neben dem Feldnamen.

Weiterhin kann das Editorbild gerollt werden, ohne daß die Feldnamen mitrollen. In diesem Fall ist die Korrespondenz auch nicht erhalten, das heißt die Inhalte stehen falsch, sind aber eigentlich richtig.

In solchen Fällen erscheint am oberen oder unteren Rand der Hinweis "Bitte ESC '1' druecken". Wenn das Editorbild nur gerollt wurde, verschwindet durch ESC '1' der Hinweis wieder und das Bild ist in Ordnung. Wenn jedoch Zeilen gelöscht oder eingefügt wurden, müssen Sie diese Änderung von Hand wieder rückgängig machen, bis der Hinweis verschwindet. Sie sollten also HOP RUBOUT und HOP RUBIN im Satzeditor nicht verwenden.

6.4 SICHERN

Wenn Sie eine Datei zum Ändern öffnen, wird von dieser Datei intern eine Sicherheitskopie angelegt, die dann geändert wird. Sie müssen diese Arbeitskopie nach den Änderungen sichern, damit die Änderungen wirksam werden.

In unserem Beispiel ist die Datei "Telefonnummern" immer noch leer. Die Änderungen sind momentan nur in der internen Kopie vorhanden. Wenn Sie die Datei zum Beispiel auf eine Archivdiskette schreiben wollten, würden Sie eine leere Datei auf der Diskette haben.

Zum Sichern rufen Sie die Funktion

- Sichern

im Hauptmenü auf. Es erscheint dann folgende Frage:

```
Datei "Telefonnummern" veraendert, sichern (j/n) ?
```

Beantworten Sie diese Frage mit 'j'. Anschließend wird Ihnen der bisherige Dateiname zum Überschreiben angeboten. Drükken Sie hier nur RETURN. Die Sicherheitskopie überschreibt dann die (leere) Version vor dem Ändern.

Alternativ könnten Sie hier auch einen anderen Namen eingeben. Dies wäre dann sinnvoll, wenn Sie den Stand vor den Änderungen noch aufbewahren wollen. In unserem Fall wäre es jedoch nutzlos, die alte leere Datei noch behalten zu wollen.

Abschließend wird gefragt, ob Sie die Arbeitskopien löschen wollen. Wenn Sie noch weiter mit der Datei arbeiten wollen, verneinen Sie diese Frage. Die Datei bleibt dann geöffnet. Anderenfalls müßten Sie die Datei neu öffnen, wenn Sie sie wieder ansehen wollen.

Wenn Sie eine neue Datei öffnen und die vorherige Datei wurde noch nicht gesichert, erscheint der gleiche Dialog. Sie können also nicht aus Versehen eine geänderte Version verlieren.

7
Drucken mit EUDAS

7.1 DRUCKMUSTER

Nachdem Sie sich die Inhalte der EUDAS-Datei ausgiebig am Bildschirm angesehen haben, möchten Sie die gespeicherten Daten sicher auch in gedruckter Form auf Papier sehen. Sie können eine EUDAS-Datei jedoch nicht ohne weiteres ausdrukken, da sie eine andere Struktur als normale Textdateien hat.

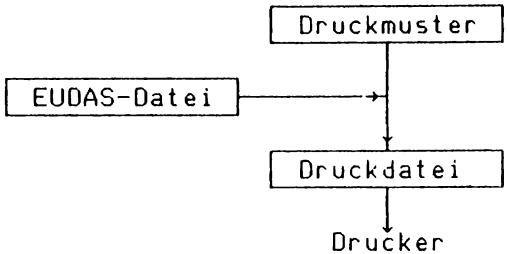
Vor dem Drucken müssen also die Inhalte der EUDAS-Datei zunächst in leserlicher Form in eine Textdatei geschrieben werden. EUDAS kann diese Aufgabe jedoch ohne Ihre Hilfe nicht alleine bewältigen.

Es taucht nämlich das Problem auf, daß die Dateiinhalte in vielen verschiedenen Formen dargestellt werden können (Sie erinnern sich sicher noch an das erste Kapitel). Vielleicht wollen Sie bestimmte Überschriften haben oder die Daten müssen auf ein bestimmtes Formular passen.

Um die Ausgabe nach Ihren Wünschen zu gestalten, müssen Sie also dem Rechner angeben, an welcher Stelle genau welche Felder gedruckt werden sollen usw. Dies geht am einfachsten, indem Sie dem Rechner ein Muster vorsetzen, nach dem er dann die richtigen Ausdrucke erstellen kann. Dieses Muster schreiben Sie in eine eigene Textdatei, die **Druckmuster** genannt wird. Aus den Daten der EUDAS-Datei und der Form, die im Druckmuster angegeben ist, wird dann eine weitere Textdatei erzeugt, die die Daten in der gewünschten Form enthält und die anschließend automatisch gedruckt werden kann.

Durch ein Druckmuster erhalten Sie fast völlige Freiheit in der Gestaltung Ihrer Ausdrucke. Zum Beispiel können aus einer einzigen Adressendatei einfache Listen, Einladungskarten oder Rundbriefe erzeugt werden. Für eine einfache Adressenliste entspricht das Druckmuster einer Zeile der Liste, wobei angegeben wird, in welche Spalten die Inhalte gedruckt werden. Zum Drucken von Einladungskarten wird als Druckmuster eine Einladungskarte verwendet, in der die Stellen markiert sind, an denen die Adresse erscheinen soll. Das gleiche kann man mit einem Brief machen, der dann mit jeder Adresse einmal ausgedruckt wird.

Man kann sich diesen Druckprozeß wie folgt vorstellen:



Vereinfacht gesagt (das genaue Verfahren wird später beschrieben) wird für jeden Satz der EUDAS-Datei das Druckmuster einmal in die Druckdatei übernommen. Dabei werden die Inhalte aus der EUDAS-Datei in einer noch anzugebenden Weise an den gewünschten Stellen eingefügt.

Im weiteren wollen wir uns damit beschäftigen, wie ein Druckmuster genau aussieht und wie daraus ein Ausdruck entsteht.

Im folgenden wollen wir zur Demonstration unsere Beispieldatei verwenden. Wenn Sie die Möglichkeit haben, sollten Sie die angegebenen Beispiele mit dieser Datei selbst am Rechner ausprobieren. Die Datei enthält folgende Felder:

"Name"
"Vorname"
"PLZ"
"Ort"
"Strasse"
"m/w"

Folgende Sätze befinden sich in der Datei:

```
Vorname Name, Strasse, PLZ Ort, m/w
-------------------------------------------------------
Herbert Wegner, Krämergasse 12, 5000 Köln, m
Helga Sandmann, Willicher Weg 109, 5300 Bonn 1, w
Albert Katani, Lindenstr. 3, 5210 Troisdorf, m
Peter Ulmen, Mozartstraße 17, 5 Köln 60, m
Karin Regmann, Grengelweg 44, 5000 Köln 90, w
Hubert Arken, Talweg 12, 5200 Siegburg, m
Anna-Maria Simmern, Platanenweg 67, 5 Köln 3, w
Angelika Kaufmann-Drescher, Hauptstr. 123, 53 Bonn 2, w
Harald Fuhrmann, Glockengasse 44, 5000 Köln 1, m
Friedrich Seefeld, Kabelgasse, 5000 Köln-Ehrenfeld, m
```

Diese Liste, in der die Felder untereinander durch Komma getrennt sind, wurde auch unter Verwendung eines Druckmusters erzeugt. Da dieses Druckmuster jedoch vorerst noch zu kompliziert ist, wollen wir erst einmal ein einfacheres Druckmuster vorstellen und versuchen, nur die Namen aus der Datei in Tabellenform auszugeben.

Das dafür nötige Druckmuster hat folgendes Aussehen:

```
% WIEDERHOLUNG
! &Name                  ! &Vorname          !
```

Das Druckmuster besteht nur aus zwei Zeilen, von der die zweite das eigentliche Muster darstellt. Die erste Zeile ist eine **Anweisung** an den Druckgenerator. "WIEDERHOLUNG" gibt an, daß die folgenden Zeilen für jeden Satz wiederholt werden sollen (warum diese Angabe notwendig ist, werden Sie später einsehen). Das Prozentzeichen kennzeichnet eine Anweisung und muß unbedingt in der ersten Spalte des Druckmusters stehen.

In der zweiten Zeile ist zu sehen, daß das Zeichen "&" dazu benutzt wird, die Stellen zu markieren, an denen nachher Feldinhalte eingesetzt werden sollen. Hinter dem "&"-Zeichen folgt der Name des Feldes, das an dieser Stelle eingesetzt werden soll. Eine solche Konstruktion wird Feldmuster genannt. Beachten Sie, daß Feldnamen hier immer ohne Anführungsstriche geschrieben werden müssen. Die Ausrufungszeichen bilden den eigentlichen Mustertext und werden unverändert als Tabellenbegrenzung in die Ausgabe übernommen.

Als Ergebnis des Druckprozesses sollte folgende Ausgabe auf dem Drucker erscheinen:

```
! Wegner              ! Herbert           !
! Sandmann            ! Helga             !
! Katani              ! Albert            !
! Ulmen               ! Peter             !
! Regmann             ! Karin             !
! Arken               ! Hubert            !
! Simmern             ! Anna-Maria        !
! Kaufmann-Drescher   ! Angelika          !
! Fuhrmann            ! Harald            !
! Seefeld             ! Friedrich         !
```

Sie können erkennen, daß die Feldmuster in der Ausgabe jeweils durch den Inhalt des zugehörigen Feldes ersetzt worden sind. Der übrige Text in der Musterzeile ist unverändert ge-

blieben. Beachten Sie, daß das "&" ein reserviertes Zeichen ist, das ein Feldmuster im umgebenden Text kennzeichnet und daher (vorerst) nicht gedruckt werden kann.

7.2 AUFRUF

Wir wollen nun besprechen, wie Sie diese Ausgabe selbst erzeugen können. Damit der Druckgenerator arbeiten kann, müssen Sie die Datei "Adressen" erst einmal öffnen. Anschließend wählen Sie das Menü

- aktuelle Datei bearbeiten

Als nächstes müssen Sie das Druckmuster erstellen. Hierfür gibt es die Auswahl

- Textdatei erstellen/aendern

da das Druckmuster eine normale Textdatei ist.

Wählen Sie diese Funktion. Sie werden dann nach einem Namen für das Druckmuster gefragt. Wir wollen das Druckmuster "Namensliste" nennen - Sie können aber auch einen beliebigen anderen Namen wählen. Denken Sie daran, die Anführungsstriche nicht mit einzugeben.

Es erscheint anschließend das Editorbild in der unteren Schirmhälfte. Geben Sie die zwei Zeilen des Druckmusters ein und beenden Sie den Editor mit ESC 'q'. Damit ist das Druckmuster fertig.

Bevor Sie jetzt weitermachen, sollten Sie überprüfen, ob an Ihrem System ein Drucker angeschlossen ist. Der Drucker sollte bei den folgenden Versuchen betriebsbereit sein.

Ist kein Drucker angeschlossen oder wollen Sie kein Papier verschwenden, haben Sie die Möglichkeit, den Ausdruck als Textdatei zu erhalten. Dieses Verfahren stellen Sie ein, indem Sie ESC ESC drücken (danach können Sie beliebige Kommandos eingeben) und das Kommando

```
direkt drucken (FALSE)
```

eingeben. Bitte die Eingabe auch hier mit RETURN abschließen.

Die Ausgaben stehen dann nachher in Dateien mit einem Namen der Form

"EUDAS-Ausgabe. n"

die Sie sich mit der oben beschriebenen Funktion dann anschauen können. Die Nummer 'n' dient zur Unterscheidung bei mehreren aufeinanderfolgenden Ausgaben. Um Verwirrung zu vermeiden, sollten Sie die Datei nach dem Anschauen löschen (im Menü "Dateiverwaltung").

Wenn Sie diese Hinweise beachtet haben, können Sier den Druckvorgang mit der Auswahl

- nach Muster drucken

starten. Sie werden hier nach dem Namen des Druckmusters gefragt, das Sie verwenden wollen (Sie können ja durchaus eine ganze Reihe von verschiedenen Druckmustern haben).

Sie können den Ablauf des Druckvorganges daran verfolgen, daß jeweils die Nummer des Satzes ausgegeben wird, der gerade bearbeitet wird. Probieren Sie eventuell auch kleine Abwandlungen des Druckmusters aus, indem Sie die Tabellenspalten schmaler oder breiter machen oder die Ausrufungszeichen durch ein anderes Zeichen ersetzen (je nach Geschmack).

Bevor der eigentliche Druckprozeß gestartet wird, wird das Druckmuster auf unsinnige oder unverständliche Konstruktionen überprüft. Ist dem Druckgenerator etwas suspekt, gibt er eine Fehlermeldung aus, in der die fragliche Situation von seiner Seite aus beschrieben wird. Er kann natürlich nicht Ihren Fehler "verstehen". Daher müssen Sie unter Umständen eine Fehlermeldung erst interpretieren, ehe Sie die wahre Ursache erkennen können.

Damit Sie einen aufgetretenen Fehler gleich korrigieren können, werden Ihnen das Druckmuster und die Fehlermeldungen parallel auf dem Bildschirm zum Ändern und Anschauen angeboten. Sie können mit dem Editor das Druckmuster ändern und in den Fehlermeldungen blättern. Diese Konfiguration wird Paralleleditor genannt (näheres dazu im EUMEL-Benutzerhandbuch unter "Fenstereditor"). Mit ESC "w" wechseln Sie zwischen den beiden Bildschirmhälften.

Wollen Sie nicht alle Namen ausdrucken, so können Sie vorher ein Suchmuster einstellen, das nur auf die gewünschten Namen zutrifft. Der Druckgenerator richtet sich immer nach dem aktuell eingestellten Suchmuster und druckt nur die ausgewählten Sätze. Wenn Sie zum Beispiel die Namen aller Frauen ausdrucken wollen, stellen Sie im Tastenmodus ein Suchmuster ein (das sollten Sie können), das für das Feld "m/w" die Bedingung "w" enthält. Danach können Sie den Druckgenerator aufrufen. Vergessen Sie nicht, das Suchmuster anschließend wieder zu löschen.

7.3 ABSCHNITTE

Die Tabellen, die Sie bis jetzt erzeugen können, sehen optisch noch nicht sehr gut aus. Es fehlt auf jeden Fall eine vernünftige Überschrift. Um eine Überschrift zu erzeugen, können Sie im Druckmuster einen **Vorspann** definieren, der ganz zu Anfang einmal gedruckt wird.

Dieser Vorspann wird durch die Anweisung

```
% VORSPANN
```

eingeleitet (bitte nicht vergessen, daß das "%"-Zeichen für eine Anweisung in der ersten Spalte stehen muß). Die folgenden Zeilen bis zur "WIEDERHOLUNG"-Anweisung gehören zum Vorspann. Ein Druckmuster für unsere Namensliste mit Überschrift könnte dann so aussehen:

```
% VORSPANN
Alle Namen aus der EUDAS-Datei "adressen"
----------------------------------------
% WIEDERHOLUNG
! &Name                ! &Vorname          !
```

Der Druckgenerator erzeugt mit diesem Druckmuster die gewünschte Liste mit Überschrift. Sie können als Vorspann natürlich auch einen beliebigen anderen Text verwenden.

In einer analogen Weise können Sie die Liste noch durch eine waagerechte Linie abschließen, indem Sie einen **Nachspann** definieren. Die dafür notwendige Anweisung heißt

```
% NACHSPANN
```

Die Zeilen nach dieser Anweisung werden gedruckt, nachdem

alle Sätze bearbeitet worden sind. Das folgende Druckmuster erzeugt schon eine sehr schöne Liste:

```
% VORSPANN
Alle Namen aus der EUDAS-Datei "adressen"
-----------------------------------------
% WIEDERHOLUNG
! &Name                ! &Vorname         !
% NACHSPANN
-----------------------------------------
```

Die drei Teile, aus denen ein Druckmuster bestehen kann (Vorspann, Nachspann und Wiederholungsteil), werden **Abschnitte** genannt. Wie Sie später noch sehen werden, haben Abschnitte eine Reihe von gemeinsamen Eigenschaften. Ein Abschnitt wird durch eine eigene Anweisung eingeleitet und endet, wenn ein anderer Abschnitt beginnt oder das Druckmuster zu Ende ist. Alle Abschnitte können auch weggelassen werden, irgendein Abschnitt muß aber immer vorhanden sein. So ist es zum Beispiel möglich, ein Druckmuster zu bauen, das nur aus einem Nachspann besteht (Sie werden allerdings jetzt noch nicht verstehen können, warum so etwas sinnvoll sein kann).

Zum Abschluß dieses Kapitels hier noch einmal eine Übersicht der bisher vorgestellten Anweisungen:

Anweisung	Bedeutung
% VORSPANN	leitet Vorspann ein
% WIEDERHOLUNG	leitet Wiederholungsteil ein
% NACHSPANN	leitet Nachspann ein

7.4 FELDMUSTER

Mit den bis jetzt beschriebenen Möglichkeiten des Druckgenerators können Sie schon sehr viel anfangen. Es fehlt aber noch die Möglichkeit, mehrere Feldinhalte direkt hintereinander zu schreiben, egal wie lang diese Inhalte sind. Diese Fähigkeit wird zum Beispiel für die anfangs vorgestellte Liste benötigt.

Die Feldmuster, die Sie bis jetzt kennen, beginnen mit einem "&"-Zeichen und werden immer genau an der Stelle gedruckt, an der sie stehen (feste Position). Sie können ein Feldmuster aber auch mit "%" beginnen lassen. In diesem Fall

kann der Inhalt verschoben werden (variable Position), je nachdem, ob vorherige Inhalte kürzer oder länger sind.

"%" ist wie "&" ein reserviertes Zeichen, kann also nicht direkt gedruckt werden. Da "&" und "%" Feldmuster einleiten, heißen sie Musterzeichen.

Um Feldmuster variabler Position einmal auszuprobieren, sollten Sie unser bisheriges Druckmuster in der folgenden Weise umschreiben:

```
% WIEDERHOLUNG
&Vorname %Name
```

(Vorspann und Nachspann der Einfachheit halber mal weggelassen). Als Ergebnis erhalten wir:

```
Herbert Hegner
Helga Sandmann
Albert Katani
Peter Ulmen
Karin Regmann
Hubert Arken
Anna-Maria Simmern
Angelika Kaufmann-Drescher
Harald Fuhrmann
Friedrich Seefeld
```

Das Feldmuster "%Name" ist also entsprechend der Länge des Vornamens nach links oder nach rechts gerutscht. Zu beachten ist, daß ein Feldmuster mit "%" nicht in der ersten Spalte stehen darf, denn dann würde die Zeile als Anweisung angesehen. Ein Feldmuster variabler Position wäre ja auch in der ersten Spalte wenig sinnvoll.

Außer den beiden bisher besprochenen einfachen Arten (mit "&" oder "%") gibt es noch weitere Ausprägungen von Feldmustern für besondere Fälle. Wird ein Feldmuster noch von weiteren Musterzeichen gefolgt, dann wird dieses Feldmuster immer in der reservierten Länge eingesetzt. Die reservierte Länge reicht vom ersten bis zum letzten Musterzeichen. Durch die zusätzlichen Musterzeichen wird also ein bestimmter Platz freigehalten.

Ersetzt man im obigen Druckmuster "&Vorname" durch "&Vorname&&", wird der Effekt des folgenden "%"-Feldes wieder aufgehoben, da jetzt für alle Vornamen die gleiche Länge verwendet wird (Probieren Sie dies aus).

Bei einem solchen Feldmuster mit fester Länge wird der Inhalt abgeschnitten, falls er diese Länge überschreitet; ist der Inhalt kürzer, wird rechts mit Leerstellen aufgefüllt. Aber auch bei Feldmustern mit variabler Länge (also ohne folgende Musterzeichen) kann abgeschnitten werden, nämlich genau dann, wenn der Inhalt so lang ist, daß ein folgendes Feld mit fester Position (mit "&" anfangend) überschrieben würde. Hätten wir also in unserem ersten Druckmuster nicht genügend Platz für die Spalten vorgesehen, wären einige Namen abgeschnitten worden (probieren Sie es nochmal aus, falls es Ihnen nicht schon passiert ist).

Wir sind jetzt aber noch nicht zu Ende mit den Feldmustervariationen. Eine letzte Möglichkeit besteht darin, den Inhalt rechtsbündig in ein Feldmuster einzusetzen. Dies hat natürlich nur Sinn bei fester Länge. Man erreicht dies dadurch, daß man das Feldmuster mit mehreren Musterzeichen beginnen läßt. So ist

```
%%Vorname%
```

die rechtsbündige Version von

```
%Vorname%%
```

Beide Feldmuster sind gleich lang, beim ersten wird jedoch am linken Rand aufgefüllt oder abgeschnitten, beim zweiten dagegen am rechten Rand.

Hier noch einmal eine Zusammenstellung aller möglichen Feldmustertypen:

Typ	Beispiel	Position	Länge	bündig
1	&Name	fest	variabel	links
2	%Name	variabel	variabel	links
3	&Name&&&	fest	fest	links
4	%Name%%%	variabel	fest	links
5	&&&Name&	fest	fest	rechts
6	%%%Name%	variabel	fest	rechts

Wir können zusammenfassen: Feldmuster dienen im Druckmuster dazu, Stellen zu markieren, an denen Inhalte eingesetzt werden sollen. Feldmuster beginnen mit einem Musterzeichen ("&" oder "%"); darauf folgt der Feldname. Durch "&" wird feste und durch "%" variable Position festgelegt. Durch zusätzliche

Musterzeichen kann eine feste Länge angegeben werden; mehrere Musterzeichen am Anfang führen zu rechtsbündigem Einsetzen.

Als nächstes wollen wir den Fall betrachten, daß Sie einen Namen in der oft auftretenden Form

```
Name, Vorname
```

schreiben wollen. Die Schwierigkeit liegt in dem Komma, das direkt hinter dem Namen folgen soll. Sie könnten versuchen, diese Situation im Druckmuster folgendermaßen darzustellen:

```
% WIEDERHOLUNG
&Name, %Vorname
```

In diesem Fall erhalten Sie aber die Fehlermeldung

```
FEHLER in Zeile 2 bei >>Name,<<
   diese Abkuerzung ist nicht definiert
```

Wenn Sie sich nicht genau vorstellen können, wie der Druckgenerator ein Feldmuster liest, wird Ihnen dieser Fehler mysteriös erscheinen, denn "Name" ist doch als Feld definiert (was eine Abkürzung ist, werden Sie später lernen). Den entscheidenden Hinweis liefert jedoch das Komma. Offensichtlich hat der Druckgenerator das Komma als Teil des Feldnamens angesehen.

Dies liegt daran, daß ja irgendwann der Feldname in einem Feldmuster beendet sein muß. Normalerweise interpretiert der Druckgenerator ein Leerzeichen oder Musterzeichen als Ende des Namens, alle vorherigen Zeichen gehören mit zum Feldnamen. Wenn nun aber nach dem Feldmuster kein Leerzeichen folgen soll (wie in unserem Beispiel) oder der Feldname selbst Leerzeichen enthält (dies ist ja erlaubt, könnte aber im Druckmuster nie erkannt werden), muß noch eine zusätzliche Angabe erfolgen.

In solchen Fällen kann der Feldname in spitze Klammern eingeschlossen werden. Der Druckgenerator sieht den Feldnamen dann bei der ersten schließenden Klammer als beendet an, wobei die Klammern natürlich nicht zum Feldnamen gehören, aber auch nicht gedruckt werden.

Das obige Beispiel müßte also richtig so formuliert werden:

```
% WIEDERHOLUNG
&<Name>, %Vorname
```

Wenn Sie dieses Druckmuster ausprobieren, werden Sie sehen, daß die Namen tatsächlich in der gewünschten Form erscheinen.

Sie sollten jetzt die Möglichkeiten des Druckgenerators soweit kennengelernt haben, daß Sie ein Druckmuster für die eingangs erwähnte Liste erstellen können. Versuchen Sie dies zunächst allein, ehe Sie die Lösung nachschauen.

Hier nun die Lösung:

```
% VORSPANN
Vorname Name, Strasse, PLZ Ort, m/w
-----------------------------------------------------
% WIEDERHOLUNG
&Vorname %<Name>, %<Strasse>, %PLZ %<Ort>, %m/w
```

Beachten Sie die spitzen Klammern, die nötig sind, um das Komma ohne Zwischenraum anzuschließen.

Als letztes Beispiel wollen wir einen Fall betrachten, bei dem pro Satz mehr als eine einzelne Listenzeile gedruckt werden soll, und zwar wollen wir einen Brief schreiben, in den der Druckgenerator die Adressen verschiedener Leute einfügen soll. Die Erstellung von Formbriefen ist eine sehr häufige Anwendung von EUDAS. Mit den bisher beschriebenen Konstrukten kann man etwa folgendes Druckmuster schreiben:

```
% WIEDERHOLUNG
     &Vorname %Name
     &Strasse
     &PLZ %Ort

     Lieber &Vorname !

     Ich lade Dich mit diesem Brief zu
     meiner nächsten Party ein.
     Bring gute Laune und was zu Essen mit.

                                Viele Grüße
#page#
```

Die letzte Zeile zeigt eine Möglichkeit, von der Sie wahrscheinlich öfter Gebrauch machen werden, nämlich Druckersteuerungsanweisungen in das Druckmuster einzufügen. Die Anweisung "#page#" wird an den Drucker weitergereicht und bewirkt, daß nach jedem Brief eine neue Seite angefangen wird (Sie wollen sicher nicht mehrere Briefe auf ein Blatt drukken). Sie können auch andere Anweisungen verwenden, z.B. neue Schrifttypen einstellen. Informieren Sie sich gegebenenfalls, welche Anweisungen die Textkosmetik zur Verfügung stellt.

Sie kennen jetzt bereits einen großen Teil der Möglichkeiten des Druckgenerators. Einige wünschenswerte Fähigkeiten fehlen jedoch noch. So wäre es vorteilhaft, wenn abhängig vom Inhalt des Feldes "m/w" die Anrede "Sehr geehrter Herr" oder "Sehr geehrte Frau" erzeugt werden könnte. Außerdem könnte das im Rechner vorhandene Datum automatisch in den Brief übernommen werden. Diese Möglichkeiten werden in weiteren Kapiteln beschrieben.

Sie sollten diese jedoch erst dann durchlesen, wenn Sie eine gewisse Sicherheit im Umgang mit Druckmustern erlangt haben. Zuvor sollten Sie die Inhalte dieses Kapitels beherrschen, damit Sie EUDAS gut nutzen können.

II.

WEITERE MÖGLICHKEITEN

8
Ketten und Koppeln

Sie haben bisher gerlernt, daß Sie immer nur eine EUDAS-Datei gleichzeitig geöffnet haben können. Sie können jedoch noch mehr Dateien öffnen. Wie dies vonstatten geht, ist der Inhalt des folgenden Kapitels.

8.1 VERKETTUNG

Wenn Sie einmal EUDAS-Dateien mit mehreren Tausend Sätzen erstellen, werden Sie feststellen, daß deren Handhabung recht umständlich sein kann. Da die Datei sehr groß ist, dauern zum Beispiel Kopiervorgänge aufs Archiv viel länger als bei kleinen Dateien.

Wenn Sie nun auch noch für jede kleine Änderung die Datei vom Archiv holen müssen und anschließend wieder zurückschreiben, werden Sie einen Weg suchen, diese Arbeit zu erleichtern. Die einfachste Möglichkeit wäre, einen schnelleren Rechner zu kaufen. Dies ist gleichzeitig aber auch die teuerste.

EUDAS ermöglicht es Ihnen nun, die große Datei in mehrere kleine aufzuspalten. In der Regel gibt es bei solch großen Dateien ein Kriterium, nach dem die Sätze in verschiedene Dateien verteilt werden können. Jede einzelne Datei kann nun bequem geändert werden.

Ein Problem entsteht jedoch, wenn alle Dateien zusammen verarbeitet werden müssen (zum Beispiel beim Drucken). Für einen solchen Vorgang können Sie die kleineren Dateien logisch aneinanderketten.

Dies bedeutet für Sie, daß alle kleinen Dateien wie eine große Datei wirken. Wenn Sie beim Bewegen in der Datei das Ende einer einzelnen Datei erreichen, kommen Sie automatisch an den Anfang der nächsten Datei.

Damit dies funktioniert, müssen die Dateien natürlich gleiche Feldstruktur haben. Außerdem ist die Zahl der verkettbaren Dateien aus technischen Gründen auf etwa 10 beschränkt.

Sie können die Dateien verketten, indem Sie bei der Funktion

- neue Datei oeffnen

die Dateiauswahl anwählen und dort die zu kettenden Dateien ankreuzen. Die erste angekreuzte Datei wird normal geöffnet, die anderen werden darangekettet. Anschließend liegen die geketteten Dateien wie eine große Datei vor. EUDAS spricht in diesem Fall von einer **virtuellen** Datei.

In der virtuellen Datei sind sowohl Einfügen als auch Ändern erlaubt. Beim Einfügen ist jedoch zu beachten, daß am Ende einer Datei nicht angefügt werden kann. Dies liegt daran, daß Sie entweder vor dem letzten Satz der einen Datei oder vor dem ersten Satz der anderen Datei einfügen. Der Endesatz der einen Datei, der normalerweise sichtbar wäre, wird übersprungen.

Beim Sichern werden Ihnen alle geketteten Dateien angeboten, wobei für jede Datei einzeln festgehalten wird, ob sie verändert wurde oder nicht. Ansonsten läuft der Dialog für jede Datei so ab, wie es bereits beschrieben wurde.

8.2 KOPPELN

Nachdem das Verketten von Dateien noch ganz einfach zu verstehen war, kommt jetzt eine Funktion auf Sie zu, die kompliziertere Möglichkeiten in sich birgt: nämlich das Koppeln.

Es kommt häufiger vor, daß sich ein Feld einer Datei auf einen bestimmten Satz in einer anderen Datei bezieht. So könnten zum Beispiel die Ausleihen einer Bücherei in folgende Datei gespeichert sein:

```
"Name"
"Vorname"
"Datum"
"Buch-Nr."
```

Wenn jetzt ein Ausleiher sein Rückgabedatum überschritten hat, möchte die Bücherei dem Kunden ein Mahnschreiben schikken. Auf diesem Schreiben soll aber nicht die Buch-Nr. erscheinen, sondern Autor und Titel des Buches.

Diese Sekundärinformationen sind in einer anderen Datei gespeichert, der Bestandskartei:

"Buch-Nr."
"Autor"
"Titel"
"Verlag"

Alle Dateistrukturen sind natürlich zwecks größerer Übersichtlichkeit vereinfacht. Um jetzt dem Kunden das Mahnschreiben zu schicken, müssen die Informationen in den beiden Dateien korreliert werden.

Dazu wird zuerst die Ausleihdatei normal geöffnet. Dazu wird dann die Bestandsdatei mit Hilfe der Funktion

- Dateien koppeln

gekoppelt. Dies hat folgenden Effekt:

Die Sätze erscheinen normal so, wie sie in der Ausleihdatei auftauchen, also für jede Ausleihe genau ein Satz. Dazu erscheint aber jeweils die Beschreibung des ausgeliehenen Buches aus der Bestandsdatei: die beiden Dateien wurden über das Feld "Buch-Nr." gekoppelt.

Als Struktur ergibt sich für die kombinierte Datei:

"Name"
"Vorname"
"Datum"
"Buch-Nr."
"Titel"
"Autor"
"Verlag"

Die Felder der Koppeldatei wurden also noch hinzugefügt.

Zwei Dinge sind in diesem Zusammenhang wichtig: Damit der Koppelvorgang ohne allzuviele Vorgaben auskommen kann, müssen Felder, über die gekoppelt wird, den gleichen Namen haben - und zwar exakt Zeichen für Zeichen. Zum zweiten muß ein solches **Koppelfeld** am Anfang der gekoppelten Datei stehen. Dies ist aus technischen Gründen notwendig, damit der Koppelvorgang in vernünftiger Geschwindigkeit ablaufen kann.

Genau wie beim Ketten ist die Kombination der Dateien nicht physikalisch, sondern nur scheinbar vollzogen worden. Die Koppeldateien gehören damit auch zur virtuellen Datei.

Nimmt man diese kurze Aussage ernst, so folgt daraus zum einen, daß Ketten und Koppeln gleichzeitig möglich sind (dem steht ja auch nichts entgegen) und daß auch mehrere Dateien gekoppelt werden können.

Letzteres könnte man in unserem Beispiel ausnutzen. Die Bücherei hat sicher auch eine Datei ihrer Mitglieder. Diese könnte etwa so aussehen:

```
"Name"
"Vorname"
"Strasse"
"PLZ"
"Ort"
```

Diese Datei können wir ebenfalls zur Ausleihdatei dazukoppeln. Damit haben wir auch gleich die Quelle gewonnen, aus der wir die Anschrift für das Mahnschreiben gewinnen können.

An dieser Stelle muß darauf hingewiesen werden, daß die Mitgliederdatei zwei Koppelfelder aufweist, nämlich "Name" und "Vorname". Damit ein Mitglied eindeutig identifiziert wird, werden beide Namen gebraucht. Dies berücksichtigt auch das Koppelverfahren. Wiederum müssen die Namen exakt mit Namen der ersten Datei übereinstimmen.

An dieser Stelle wollen wir kurz als Regel aufstellen, unter welchen Bedingungen Dateien gekoppelt werden können.

Die ersten Felder der Koppeldatei, die wörtlich auch in der ersten Datei auftauchen, werden Koppelfelder genannt. Zu einem Satz der ersten Datei wird ein Satz der Koppeldatei gezeigt, der im Inhalt der Koppelfelder übereinstimmt.

Übersetzt in unser Beispiel heißt dies: "Buch-Nr." bzw. "Name" und "Vorname" sind Koppelfelder. Zu einer bestimmten Ausleihe erscheinen die Daten des Buches mit der angegebenen Buch-Nr. bzw. die Adresse des Mitgliedes mit den angegebenen Namen.

8.3 AUSWIRKUNGEN DES KOPPELNS

Nachdem Sie nun das Grundprinzip des Koppelns kennen, wollen wir einige Auswirkungen dieses Verfahrens kennenlernen.

Zuerst muß geklärt werden, was passiert, wenn es keinen passenden Satz in der Koppeldatei gibt. Zum Beispiel könnte eine Buch-Nr. eingegeben worden sein, die in der Bestandsdatei nicht existiert. In diesem Fall zeigt EUDAS für die Felder der Koppeldatei einfach einen leeren Inhalt an. Wenn umgekehrt zu einem bestimmten Buch keine Ausleihe existiert, macht das natürlich nichts - das Buch erscheint nur dann, wenn Sie die Bestandsdatei alleine öffnen.

Weiterhin kann es passieren, daß es zwei passende Sätze in der Koppeldatei gibt. Dies kommt dann vor, wenn zwei Mitglieder gleichen Namen und gleichen Vornamen haben (Was gar nicht so selten ist). In diesem Fall zeigt EUDAS beide Kombinationen an. Die Ausleihe erscheint also zweimal, jeweils mit einem anderen Mitglied.

Damit man diesen Fall ohne weiteres erkennen kann, führt EUDAS bei Kopplungen zwei Nummern: zum einen die normale Satznummer und zum anderen eine Kombinationsnummer. In dem eben besprochenen Fall würde die Satznummer gleichbleiben, die Kombinationsnummer aber hochgezählt werden.

Dieses Durchzählen aller Kombination passiert aber nur dann, wenn Sie sich mit der Funktion "Satz weiter" in der Datei bewegen. Wenn Sie rückwärts gehen oder auf einen bestimmten Satz positionieren, wird immer nur die erste Kombination angezeigt (Dies hat zum Teil technische Gründe).

Wenn Sie Dateien gekoppelt haben, können Sie immer noch ändern (entsprechende Angabe vorausgesetzt). Die Auswirkungen der Veränderungen sind jedoch nicht mehr ganz so einfach wie bei geketteten Dateien, wo sich die Änderungen ja einfach auf den aktuellen Satz bezogen.

Als Grundregel gilt, daß Änderungen nach Möglichkeit keine Auswirkungen auf die Koppeldateien haben sollen. Das führt dazu, daß beim Einfügen eines neuen Satzes (oder beim Tragen) die Operation nicht in der Koppeldatei durchgeführt wird. Dies ist auch nicht nötig, denn wenn ein neuer Satz eingefügt wird, existiert sowieso noch kein passender Satz in der Koppeldatei und die entsprechenden Felder bleiben leer.

Änderungen an den Koppelfeldern können nun zu drei verschiedenen Reaktionen führen:

1. Es wird kein Satz der Koppeldatei geändert, sondern nur ein neuer passender Satz gesucht. Dies geschieht immer dann, wenn außer den Koppelfeldern nur leere Inhalte für die Felder der Koppeldatei angegeben sind. Nach dem

Ändern oder Einfügen werden dann die Inhalte des neuen Koppelsatzes angezeigt.

2. Es wird ein neuer Satz in der Koppeldatei angefügt. Dies geschieht immer dann, wenn die Koppelfelder verändert wurden und die anderen Felder der Koppeldatei nicht leer sind. Dadurch soll verhindert werden, daß die Koppelfelder in einem Satz verändert werden, der vielleicht noch zu einem anderen Satz paßt.

3. Der Satz in der Koppeldatei wird verändert. Dies geschieht nur dann, wenn die Koppelfelder unverändert geblieben sind, der Rest sich aber geändert hat.

Da Koppeldateien keine Ordnung besitzen müssen, werden neue Sätze der Koppeldatei immer am Ende angefügt. Dies ist zu beachten, wenn die Koppeldatei auch allein verwendet werden soll.

9
Möglichkeiten zur Auswahl

In diesem Kapitel wollen wir weitere Möglichkeiten des Suchmusters erklären. Außerdem wird eine Methode zum manuellen Auswählen von Sätzen vorgestellt.

9.1 ALTERNATIVEN

Die bisher genannten Möglichkeiten des Suchmusters sind noch etwas beschränkt. Eine Bedingung in unserer Adressendatei, die wir im Suchmuster noch nicht ausdrücken können, wäre zum Beispiel: Suche alle Adressen der Personen, die Wegner oder Simmern heißen.

Diese Alternative, Wegner ODER Simmern, kann nun in EUDAS durch ein Komma ausgedrückt werden:

```
Name      Wegner,Simmern
Vorname
```

Beachten Sie, daß hinter dem Komma kein Leerzeichen folgen darf, wie Sie es vielleicht gewohnt sind in einem Text zu schreiben. EUDAS kann nämlich nicht unterscheiden, ob Sie das Leerzeichen nur aus optischen Gründen geschrieben haben, oder ob Sie danach suchen wollen.

Die eben beschriebene Konstruktionsmethode heißt lokale **Alternative**. Lokal deshalb, weil Sie nur innerhalb eines Feldes gilt. Was das bedeuten soll, sehen Sie, wenn wir die Bedingung mit einer weiteren Bedingung für ein anderes Feld kombinieren:

```
Name      Wegner,Simmern
Vorname
Strasse
PLZ       5*
Ort
```

Dieses Muster hat die Bedeutung: Wähle alle Personen namens Wegner oder Simmern aus, die im PLZ-Bereich 5 wohnen. Die bei-

den Bedingungen für den Namen sind mit der Bedingung für die PLZ mit UND verknüpft - das heißt, eine der beiden ersten Bedingungen muß zutreffen UND die untere Bedingung. Dieses UND ist global, da es Bedingungen für verschiedene Felder miteinander verbindet.

Natürlich können Sie für mehrere Felder gleichzeitig lokale Alternativen angeben. Eine anderes Suchmuster könnte zum Beispiel so aussehen:

```
Name      Wegner,Simmern
Vorname
Strasse
PLZ       5,5000
Ort
```

In diesem Fall muß eine ausgewählte Person Wegner oder Simmern heißen und in Köln wohnen.

Es wird nun aber für bestimmte Situationen noch eine andere Art von Alternativen benötigt. Als Beispiel wollen wir ein Suchmuster angeben, das folgende Bedingung ausdrückt. Gesucht ist eine weibliche Person mit Namen Simmern oder eine männliche Person mit Namen Wegner.

Dieser Fall läßt sich mit unseren bisherigen Mitteln nicht lösen. Es wird nämlich eine Alternative zwischen zwei zusammengesetzten Bedingungen gefordert. Als Ausweg bietet sich an, praktisch mehrere Suchmuster anzugeben, die dann mit ODER verknüpft werden.

Um diese verschiedenen Suchmuster optisch am Bildschirm zu kennzeichnen, wird ein Semikolon verwendet. Das Semikolon trennt das Suchmuster quasi in verschiedene Spalten auf, die jeweils eine eigene Bedingung enthalten können. Unser gewünschtes Suchmuster würde also so aussehen:

```
Name      Wegner;Simmern
Vorname
Strasse
PLZ
Ort
m/W       m;W
```

Ebenso wie bei lokalen Alternativen darf hinter dem Semikolon kein Leerzeichen folgen. Daher kann das zweite Semikolon auch nicht direkt unter dem ersten stehen. Die Spalten werden also einfach nur durchgezählt: nach dem ersten Semikolon

beginnt die zweite Spalte, nach dem zweiten Semikolon die dritte usw.

In Zeilen, in denen keine Bedingungen stehen, kann auch das Semikolon weggelassen werden. Es kann ebenfalls weggelassen werden, wenn die weiteren Spalten leer sind. Steht ein Semikolon direkt am Anfang der Zeile, so ist die erste Spalte leer.

Um dies zu illustrieren, sei hier noch ein weiteres Beispiel für die sogenannte **globale Alternative** angegeben:

```
Name      Wegner
Vorname   ;Anna-Maria
Strasse
```

In diesem Fall muß eine ausgewählte Person mit Nachnamen Wegner oder mit Vornamen Anna-Maria heißen.

9.2 NEUE BEDINGUNGEN

Im vorigen Abschnitt haben wir zur Vereinfachung nur Bedingungen betrachtet, die exakt zutreffen mußten. Sie wissen aber bereits, daß man auch Bedingungen angeben kann, bei denen nur ein Teil des zu suchenden Feldes bekannt ist, nämlich indem der unbekannte Teil mit einem Stern markiert wird.

In Kapitel 5 haben Sie gelernt, daß der Stern nur am Anfang und Ende des Musters stehen kann. Dies trifft nicht ganz zu, denn Sie können den Stern auch inmitten eines Textes anwenden. So trifft die Bedingung "Si*n" auf alle Namen zu, die mit "Si" beginnen und mit "n" enden.

Beachten Sie hier das "und" in der Formulierung der Bedingung. Das Muster ist eigentlich eine Schreibweise für zwei Bedingungen für ein Feld, die mit UND verknüpft sind.

Sie können auch noch weitere Sterne in das Muster aufnehmen. Dabei gibt es jedoch eine Kleinigkeit zu beachten. Das Muster "*x*y*" bdeutet: das Feld muß ein "x" und ein "y" enthalten. Über die Reihenfolge der beiden Zeichen ist jedoch in dieser Bedingung nichts gesagt, obwohl es vielleicht vom Aussehen suggeriert wird.

Denken Sie daran, keine zwei Sterne nebeneinander zu schreiben - eine solche Bedingung hätte keinen Sinn.

Es gibt eine weitere spezielle Bedingung, die mit Hilfe des Sterns formuliert wird. Ein einzelner Stern bedeutet nämlich: Das Feld ist nicht leer. Beachten Sie den kleinen Unterschied: ein Stern in einem Muster kann für einen beliebigen

Text stehen, der auch leer sein kann. Ein einzelner Stern jedoch steht für einen beliebigen Text, der nicht leer ist.

Damit Sie ein Gefühl für die Verwendung des Sterns bekommen, hier noch ein paar Beispiele:

Mei*r*

Der Name beginnt mit "Mei" und enthält ein "r". Trifft zu auf "Meier", "Meiring", aber nicht auf "Meiling" oder "Meizer".

Donau*dampf*schiff*schaft

Feld beginnt mit "Donau", endet mit "schafft" und enthält "dampf" und "schiff". Trifft zu auf "Donaudampfschiffahrtsgesellschaft", aber auch auf "Donaugesellschiffdampffahrtschaft".

Roller*erfahren

Dieses Muster muß man ganz genau interpretieren. Es bedeutet: der Inhalt beginnt mit "Roller" und endet mit "erfahren". Das Muster trifft nicht nur auf "Roller erfahren" sondern auch auf "Rollerfahren" zu. Der Stern verliert also in diesem Fall seine symbolische Bedeutung als Platzhalter für einen bestimmten Text.

Es gibt in EUDAS noch weitere Muster, die einen ganzen Bereich von Werten auswählen. Diese betreffen Bedingungen der Art "größer als" und "kleiner als". Solche Vergleichsbeziehungen werden durch zwei Punkte dargestellt.

So wählt das Muster "K.." alle Felder aus, die in der alphabetischen Reihenfolge hinter "K" liegen, wobei das "K" selbst mit eingeschlossen ist. Umgekehrt trifft "..K" auf alle Felder zu, die davor liegen.

Sie können beide Bedingungen auch kombinieren. So trifft die Bedingung "A..K" auf alle Felder zu, die im Lexikon unter "A" bis "L" erscheinen (die Felder mit "K" sind hier ausgeschlossen). Beachten Sie, daß die direkte Kombination wieder die Verknüpfung zweier einzelner Bedingungen mit UND darstellt.

Um den Bereich möglicher Suchmuster noch zu erweitern, können Sie einzelne Bedingungen auch noch verneinen. Dies geschieht durch Voranstellen zweier Minuszeichen. So bedeutet das Muster "--Meier", daß alle Personen ausgewählt werden, die nicht Meier heißen.

Die Verneinung bezieht sich immer auf das unmittelbar folgende Muster und nicht etwa auf eine ganze Zeile. Sie umfaßt jedoch die UND-Verknüpfung der kombinierten Bedingungen. So sind zum Beispiel die Muster "--E..M" und "--E..,--..M" und "..E,M.." völlig gleichbedeutend.

Hier noch eine Bemerkung zur Geschwindigkeit des Suchens. Je mehr Bedingungen Sie angeben, desto mehr Vergleiche müssen beim Suchen angestellt werden und desto länger dauert es.

Das erste Feld einer Datei erfährt jedoch eine Sonderbehandlung. Wenn Sie für dieses Feld ein Muster für Übereinstimmung angeben, kann der Suchvorgang enorm beschleunigt werden, da das erste Feld einer Datei intern speziell verwaltet wird. Damit das Verfahren funktioniert, dürfen keine globalen Alternativen oder lokale Alternativen für das erste Feld verwendet werden.

Diese Suchoptimierung sollten Sie bereits beim Einrichten einer Datei berücksichtigen. Geben Sie als erstes Feld das an, nach dem am ehesten direkt gesucht wird. Typisches Beispiel hierfür ist der Nachname, aber auch Artikelnummern sind sinnvoll. Wichtig ist, daß das erste Feld nicht zu oft identisch ist und auch mehr als zwei Buchstaben enthält, damit die Optimierung ihre volle Wirksamkeit entfaltet.

9.3 MARKIERUNG

Manchmal entsteht die Situation, daß Sie eine Reihe von Sätzen bearbeiten wollen, aber keine Suchbedingung formulieren können, die auf alle diese Sätze zutrifft. In diesem Fall bietet EUDAS Ihnen die Möglichkeit, solche Sätze von Hand zu markieren.

Beispiel: Sie haben eine ganze Reihe von Sätzen geändert und wollen diese Änderungen als Protokoll ausdrucken. Es läßt sich aber nicht mit Hilfe eines Suchmusters feststellen, welche Sätze geändert wurden.

Als Abhilfe wählen Sie bei jedem geänderten Satz die Funktion

- Markierung aendern

Dadurch wird der bisher unmarkierte Satz markiert. Dies wird kenntlich an der Anzeige "MARK+" in der Überschrift. Sobald Sie den ersten Satz markiert haben, erscheint bei jedem Satz, ob er markiert ist oder nicht.

Haben Sie einen Satz irrtümlich markiert, können Sie die Markierung mit der gleichen Funktion auch wieder entfernen.

Alle Funktionen, die bisher die durch das Suchmuster ausgewählten Sätze bearbeitet haben, arbeiten nun nur noch auf den markierten Sätzen. Somit können Sie anschließend mit der Druckfunktion die gewünschten Sätze drucken. Lediglich die Bewegung am Bildschirm beachtet immer nur die Suchbedingung.

Sie können alle Markierungen der Datei mit der Funktion

- Markierungen loeschen

wieder entfernen. Die Markierungen verschwinden auch, wenn eine neue Datei geöffnet wird. Die Markierungen sind also nicht permanent in einer Datei gespeichert, sondern existieren nur, während die Datei geöffnet ist.

Bei Koppeldateien können Sie aus technischen Gründen immer nur alle Kombinationen auf einmal markieren. Die Markierung einer Kombination markiert auch alle anderen Kombinationen des gleichen Satzes.

10
Behandlung von EUDAS-Dateien

In diesem Kapitel wird beschrieben, wie EUDAS-Dateien sortiert werden können. Außerdem wird erklärt, was Reorganisieren bedeutet.

10.1 SORTIEREN

Wenn Sie die Sätze in Ihrer EUDAS-Datei in einer bestimmten Reihenfolge haben wollen (dies wird in vielen Fällen zum Drucken verlangt), müssen Sie die Datei sortieren. Sie können EUDAS angeben, in welcher Reihenfolge die Sortierung erfolgen soll.

Um eine Datei zu sortieren, rufen Sie die Funktion

- EUDAS-Datei sortieren

auf. EUDAS fragt Sie dann nach dem Dateinamen. Falls die Datei noch nie sortiert wurde, wird Ihnen auf jeden Fall die Sortierreihenfolge zum Auswählen angeboten. Anderenfalls werden Sie gefragt, ob Sie die vorherige Sortierreihenfolge ändern wollen.

Die Sortierreihenfolge gibt an, welche Felder in welcher Reihenfolge beim Vergleichen zweier Sätze benutzt werden sollen. Zuerst wird das an erster Stelle angegebene Feld verglichen. Sind die Inhalte hier unterschiedlich, wird die Einordnung der Sätze nach diesem Feld bestimmt.

Sind die Inhalte in diesem Feld aber gleich, so wird nach dem nächsten Feld verglichen. Ist kein weiteres Feld in der Sortierreihenfolge angegeben, wird der Vergleich an dieser Stelle mit einem zufälligen Ergebnis abgebrochen, das heißt, es kann nicht vorhergesagt werden, welcher der beiden Sätze zuerst kommt.

Die Sortierreihenfolge können Sie in einer Menüauswahl eingeben. Kreuzen Sie die Felder an, die Sie vergleichen wollen und achten Sie auf die richtige Reihenfolge. Die eingegebene Reihenfolge wird in der Datei gespeichert, um beim nächsten Sortiervorgang wiederverwendet zu werden.

Der Ablauf des Sortierens wird durch Ausgabe von Satznummern dargestellt. Bis zur ausgegebenen Satznummer sind alle Sätze richtig sortiert. Bei Bedarf kann der Vorgang durch SV und dann 'halt' abgebrochen werden. Die Datei bleibt dabei auf jeden Fall intakt.

Die gespeicherte Sortierreihenfolge wird auch noch zu einer weiteren Optimierung benutzt. Wenn eine Datei sortiert war und nur wenige Änderungen stattgefunden haben, brauchen beim nächsten Sortiervorgang nur die wenigen veränderten Sätze einzeln einsortiert zu werden. Das funktioniert natürlich nur unter der Voraussetzung, daß die gleiche Sortierreihenfolge gewählt wird. Das Sortieren braucht in diesem Fall erheblich weniger Zeit.

10.2 FELDTYPEN

Normalerweise werden die einzelnen Felder nach dem EUMEL-Zeichencode verglichen. Das bedeutet, daß sich die Reihenfolge der Zeichen nach dem EUMEL-Zeichencode richtet. Ein Zeichen mit einem höheren Code wird also vor einem Zeichen mit einem niedrigeren Code einsortiert.

In manchen Fällen ergeben sich mit diesem Vergleichsverfahren aber auch Schwierigkeiten. Wenn in einem Feld Zahlen oder DM-Beträge stehen, führt die Methode zu falschen Ergebnissen. Die "10" wird zum Beispiel vor der "2" einsortiert. Warum? Texte werden immer linksbündig geschrieben und verglichen. Bei Zahlen richtet sich die Wertigkeit jedoch nach dem Abstand vom Komma.

Da bei Texten zuerst das erste Zeichen verglichen wird, ensteht hier durch Vergleich von "1" und "2" der Eindruck, die "10" käme vor der "2". Korrigieren könnte man dies, indem man ein Leerzeichen vor die "2" schreibt. Wenn also die (nicht geschriebenen) Dezimalkommata direkt untereinanderstehen, werden Zahlen richtig verglichen.

EUDAS hat jedoch eine bequemere Art, dieses Problem zu behandeln. Ein Feld, das Zahlen enthalten soll, bekommt einen speziellen Typ ZAHL zugewiesen, der zu einer richtigen Sortierung führt.

Bei Feldern vom Typ ZAHL ignoriert EUDAS alle nichtnumerischen Zeichen und vergleicht den Wert der Zahl. So können Sie zum Beispiel in einem Satz "2,50 DM" und im anderen Satz

"10 DM" eintragen - EUDAS kann jetzt die richtige Reihenfolge feststellen.

Übrigens: falls Sie numerische Werte lieber mit einem Dezimalpunkt statt einem Dezimalkomma schreiben, können Sie EUDAS das mit Hilfe des Aufrufs

```
dezimalkomma (".")
```

mitteilen. Drücken Sie ESC ESC, um dieses Kommando im Menü einzugeben.

Feldtypen können Sie in der Funktion

- Feldstruktur aendern

angeben. Dazu werden Ihnen bei der Auswahl zu ändernder Felder alle Feldnamen und die zugehörigen Feldtypen angezeigt. Kreuzen Sie hier die Felder an, deren Feldtypen Sie ändern möchten. Sie können dann bei diesen Feldern den Namen überschreiben. Geben Sie aber im Normalfall ein RETURN ein, damit der Name unverändert bleibt.

Anschließend können Sie für das Feld den neuen Feldtyp angeben. Tippen Sie einen der vier Feldtypen als Text ein und drücken Sie RETURN. Anschließend hat das Feld einen neuen Typ. Die verschiedenen möglichen Typen werden jetzt erklärt.

TEXT ist der Standardtyp, der die Feldinhalte nach EUMEL-Zeichencode vergleicht. Den Typ ZAHL hatten wir schon weiter oben kennengelernt. Zusätzlich gibt es noch einen Typ DATUM.

Dieser Typ vergleicht Daten der Form "tt.mm.jj". Soll ein solches Datum richtig einsortiert werden, müßte es anderenfalls in der Reihenfolge umgedreht werden (also "jj.mm.tt"). Dies ist aber nicht nötig, wenn das Feld den Typ DATUM bekommt.

Der letzte Typ ist DIN. Dabei werden Texte nach DIN 5007 verglichen. Das bedeutet, daß Groß- und Kleinbuchstaben als gleich angesehen werden, daß alle nichtalphabetischen Zeichen ignoriert werden und die Umlaute ihren richtigen Platz bekommen (Umlaute werden in normalen Texten hinter allen anderen Zeichen einsortiert). Da hierfür ein relativ großer Zeitaufwand notwendig ist, sollte dieser Typ nur dann gewählt werden, wenn er erforderlich ist. Den schnellsten Vergleich ermöglicht der Typ TEXT.

Beachten Sie, daß mit der Vergabe von Feldtypen keine Überprüfung der Eingabe verbunden ist. Insbesondere beim Datum wird nicht geprüft, ob die Form "tt.mm.jj" eingehalten wurde. Wollen Sie solche Überprüfungen vornehmen, lesen Sie bitte Kapitel 12.

10.3 REORGANISIEREN

Wenn eine Datei viel geändert wurde, führen zwei Effekte zu einer langsameren Verarbeitung dieser Datei. Zum einen wird durch Textleichen der Platzbedarf größer. Dies tritt vor allem dann auf, wenn zu einzelnen Sätzen immer etwas hinzugefügt wurde.

Da der Platzbedarf der Datei also wächst, sind mehr Speicherzugriffe notwendig, als es dem Inhalt entspricht. Doch nicht nur der Platz, sondern auch die Verteilung der Sätze machen sich unangenehm bemerkbar. Da vergrößerte Sätze intern am Ende der Datei gespeichert werden, werden logisch aufeinanderfolgende Sätze physikalisch weit verstreut.

Der gleiche Effekt ensteht auch durch Umsortieren oder Einfügen von Sätzen. Um die Datei sequentiell zu bearbeiten, sind also ständig wechselnde Speicherzugriffe erforderlich.

Die beiden beschriebenen Effekte führen zur Geschwindigkeitsverringerung. Dies kann verhindert werden, indem die Datei in eine frische Datei umkopiert wird. Diesen Vorgang nennt man **Reorganisieren**. Um diese Funktion durchzuführen, rufen Sie

- Datei reorganisieren

auf. Während des Umkopierens werden die Satznummern ausgegeben. Achten Sie darauf, daß zum Reorganisieren genügend Platz auf dem System vorhanden ist, um eine komplette Kopie der zu reorganisierenden Datei aufzunehmen.

11
Variationen der Anzeige

In diesem Kapitel soll beschrieben werden, welche Möglichkeiten Sie noch bei der Anzeige und Änderung von EUDAS-Dateien haben.

11.1 ROLLEN

Da die Anzahl der möglichen Felder bei EUDAS-Dateien viel größer ist als die Anzahl der zur Verfügung stehenden Zeilen auf dem Bildschirm (255 gegenüber 16), muß es eine Möglichkeit geben, auch die übrigen Felder anzusehen.

Dazu kann man den Bildschirmausschnitt in vertikaler Richtung rollen. Sie müssen sich die Bildschirmanzeige als einen Ausschnitt des wirklichen Satzes vorstellen. Um weitere Inhalte des Satzes zu sehen, verschieben Sie den Ausschnitt einfach.

Dies geschieht mit Hilfe der Funktion

- Bild vertikal verschieben

Sie gelangen dabei in einen Rollzustand, in dem Sie Tasten zum Verschieben des Ausschnitts drücken können. Mit den Kombinationen ESC UNTEN und ESC OBEN können Sie den Ausschnitt um eine Seite in die gewünschte Richtung verschieben. Dies funktioniert jedoch nur, wenn auch noch weitere Inhalte vorhanden sind, die gezeigt werden können.

Mit den Pfeiltasten UNTEN und OBEN können Sie das Bild jeweils um eine Zeile verschieben. Sie können den Rollzustand mit ESC 'q' wieder verlassen.

Sie können auch Rollen, wenn Sie sich im Satzeditor befinden (also beim Suchen, Einfügen und Ändern). In diesem Fall wirken UNTEN und OBEN jedoch direkt auf den Cursor. Achten Sie darauf, daß der Cursor so nicht aus dem Schreibfeld bewegt wird, da sonst die Anzeige in Unordnung gerät (s. Kapitel 6).

Das Rollen ist oft auch dann notwendig, wenn sich ein Feld über mehrere Zeilen erstreckt. In diesem Fall kann es passieren, daß die erste Zeile des Feldes nicht mehr sichtbar ist, da sie oberhalb des Anzeigebereichs liegen würde.

11.2 FELDAUSWAHL

Eine weitere Möglichkeit bei zu vielen Feldern besteht darin, nur die interessanten Felder zur Anzeige auszuwählen. Dies geschieht mit der Funktion

- Feldauswahl

Ihnen werden alle Felder zur Auswahl angeboten. Kreuzen Sie die Felder an, die Sie sehen wollen und denken Sie daran, daß die Reihenfolge des Ankreuzens beachtet wird. Anschließend werden Ihnen nur die ausgewählten Felder angezeigt.

Beachten Sie, daß die Auswahl der anzuzeigenden Felder nichts mit der eigentlichen Dateistruktur zu tun hat, sondern nur für die Anzeige gilt. Den Verarbeitungsfunktionen (zum Beispiel Drucken) stehen natürlich nach wie vor alle Felder zur Verfügung.

Unvermutete Effekte können dann entstehen, wenn Sie bei einer eingestellten Feldauswahl ändern oder einfügen. Die nicht angezeigten Felder werden beim Ändern natürlich nicht geändert und beim Einfügen einfach leer gelassen.

Wollen Sie die Anzeige aller Felder wiederherstellen, so brauchen Sie nicht alle Felder mühsam einzeln anzukreuzen. Mit HOP 'x' werden gleich alle Felder auf einmal angekreuzt.

11.3 ÜBERSICHT

Wie Sie bisher gesehen haben, zeigte EUDAS immer einen einzigen Satz in dem Standardformular auf dem Bildschirm. Es gibt jedoch auch eine Möglichkeit, mehrere Sätze gleichzeitig zu betrachten. Dazu dient die **Übersichtsanzeige**.

In der Übersichtsanzeige nimmt jeder Satz nur eine Bildschirmzeile in Anspruch. Die Feldinhalte werden, durch Komma getrennt, in der Zeile nacheinander aufgezählt, bis kein Platz mehr vorhanden ist. Am Anfang jeder Zeile steht die Satznummer und ob der jeweilige Satz markiert ist. In der Überschrift stehen in gleicher Weise die Feldnamen angegeben.

```
Satznr. Name, Vorname, PLZ, Ort, Strasse, m/w,
1     - Wegner, Herbert, Krämergasse 12, 5000, Köln, m,
2     - Sandmann, Helga, Willicher Weg 109, 5300, Bonn 1, w,
3     - Katani, Albert, Lindenstr. 3, 5210, Troisdorf, m,
4     - Ulmen, Peter, Mozartstraße 17, 5, Köln 60, m,
5     - Regmann, Karin, Grengelweg 44, 5000, Köln 90, w,
6     - Arken, Hubert, Talweg 12, 5200, Siegburg, m,
7     - Simmern, Anna-Maria, Platanenweg 67, 5, Köln 3, w,
8     - Kaufmann-Drescher, Angelika, Hauptstr. 123, 53, Bonn 2, w,
9     - Fuhrmann, Harald, Glockengasse 44, 5000, Köln 1, m,
10    - Seefeld, Friedrich, Kabelgasse, 5000, Köln-Ehrenfeld, m,
11    - << DATEIENDE >>
```

Abb. 11-1 Übersichtsanzeige

Der aktuelle Satz wird innerhalb der Übersichtsanzeige immer durch eine inverse Satznummer dargestellt. Es werden nur jeweils ausgewählte Sätze gezeigt. Trifft die Selektionsbedingung nicht auf den aktuellen Satz zu, wird an seiner Stelle zur Information ein leerer Platzhalter angezeigt. Hinter dem letzten Satz wird auch das Dateiende als besonders gekennzeichneter Satz angegeben.

Die Umschaltung zwischen Normaldarstellung und der Übersichtsanzeige geschieht mit Hilfe der Funktion

- Darstellung umschalten

Es wird die jeweils andere Darstellungsart gewählt.

Auch in der Übersichtsanzeige ist Rollen möglich. In diesem Fall findet eine Bewegung innerhalb der Datei statt, der aktuelle Satz wird also geändert. Wird innerhalb der Übersichtsanzeige der Satzeditor aufgerufen (also Suchen, Einfügen oder Ändern), wird automatisch zeitweise in die Normaldarstellung umgeschaltet.

In der Übersichtsanzeige ist eine getrennte Feldauswahl möglich. Das heißt, die Feldauswahl wirkt immer nur auf die aktuelle Darstellungsweise. So können Sie die Übersichtsanzeige zum Überblick über verschiedene Felder benutzen, während Sie beim Ändern andere Felder angezeigt bekommen.

Insbesondere in der Übersichtsanzeige ist die Feldauswahl sinnvoll, da so nur die wirklich interessanten Inhalte angezeigt werden und alles Wichtige auch in eine Zeile paßt.

12
Tragen

In diesem Kapitel lernen Sie, wie man komplette Sätze von einer EUDAS-Datei in die andere trägt. Dabei sind auch Überprüfungen des Inhalts möglich.

12.1 EINZELSATZ

Vielleicht haben Sie inzwischen schon nach einem Kommando gesucht, um Sätze zu löschen. EUDAS verwendet hierfür jedoch ein anderes Konzept. Sollen Sätze aus einer Datei entfernt werden, müssen Sie in eine andere Datei getragen werden. Diese Datei funktioniert dann gewissermaßen als Mülleimer.

Dies hat den Vorteil, daß Ihre Daten nicht sofort verschwinden, sondern noch wiedergeholt werden können. Zum anderen ist ein solches Verfahren auch bei vielen Dokumenten erforderlich, wo es ja gewisse Aufbewahrungsfristen gibt.

Wenn Sie die Sätze dann tatsächlich löschen wollen, müssen Sie quasi den Mülleimer ausleeren, also die Zieldatei löschen. Damit sind die Informationen dann endgültig weg.

Sie können den aktuellen Satz in eine andere Datei tragen, indem Sie die Funktion

- Satz tragen

aufrufen. Sie werden dann nach dem Namen der Zieldatei gefragt. Hier können Sie zum Beispiel "müll" angeben.

Existiert die Zieldatei noch nicht, wird sie nach Anfrage eingerichtet. Existiert sie jedoch schon, muß Sie mindestens so viele Felder wie die aktuelle Datei haben, damit keine Informationen verlorengehen können.

Der getragene Satz wird in der Zieldatei am Ende angefügt und in der aktuellen Datei gelöscht. Beachten Sie jedoch, daß bei gekoppelten Dateien der Satz in der Koppeldatei nicht gelöscht wird. Sie können also aus Koppeldateien keine Sätze entfernen.

Umgekehrt können Sie einen Satz auch einfach wieder aus dem Mülleimer herausholen. Dazu dient die Funktion

- Satz holen

Diese holt den letzten Satz der Zieldatei und fügt ihn wieder vor dem aktuellen Satz ein. Tragen und Holen sind also spiegelbildliche Operationen. Der geholte Satz wird natürlich in der Herkunftsdatei gelöscht.

Auch in diesem Fall müssen die Felderzahlen der beiden Dateien gleich sein. Wenn kein Satz in der Zieldatei vorhanden ist, wird eine Fehlermeldung ausgeben, ebenso, wenn die angegebene Datei nicht existiert.

Denken Sie auch daran, daß Sie sich bei der Eingabe des Dateinamens eine Auswahl abrufen können (mit ESC 'z').

Die beiden angegebenen Operationen lassen sich auch dazu verwenden, Sätze von Hand an eine andere Stelle in der Datei zu bringen. Dazu tragen Sie den Satz zuerst in eine Zwischendatei und holen ihn dann wieder an die richtige Stelle.

12.2 TRAGEN MIT PRÜFEN

Neben den eben beschriebenen Funktionen haben Sie auch noch die Möglichkeit, den ausgewählten Teil der aktuellen Datei zu tragen. Dies dient zum einen zum Entfernen von markierten Sätzen, zum ändern können dabei auch Bedingungen überprüft werden.

Diese **Prüfbedingungen** sollen sicherstellen, daß die Daten in einer Datei ganz bestimmten Richtlinien entsprechen. Zum Beispiel kann geprüft werden, ob ein eingegebenen Datum stimmen kann, ob ein Satz doppelt aufgenommen wurde oder ob eine Artikelnummer die richtige Anzahl von Stellen hat.

Die Prüfbedingungen werden einer Datei fest zugeordnet. Sie können bei der Funktion

- Feldstruktur aendern

angegeben werden, wenn Sie dort die letzte Frage bejahen. Sie geben dann die einzelnen Prüfbedingungen ein.

Das ganze Verfahren läuft nun so ab: Sie tragen die neuen Sätze in eine Zwischendatei ein, die die gleiche Struktur wie die eigentliche Datei hat. Wenn Sie alle Sätze eingegeben haben, tragen Sie diese Datei komplett in die gewünschte Datei. Dabei werden die Prüfbedingungen getestet.

Erfüllt ein Satz die Bedingungen, wird er anstandslos getragen. Trifft eine Bedingung aber nicht zu, bleibt der Satz in der Zwischendatei und eine entsprechende Meldung wird ausgegeben. Die Meldungen werden gespeichert, um Sie später nochmal abrufen zu können.

Sie müssen jetzt in der Zwischendatei die notwendigen Änderungen durchführen. Beim Aufruf der Funktion

- Satz aendern

bekommen Sie jetzt automatisch im oberen Bildschirmteil die diesen Satz betreffenden Meldungen gezeigt. Anhand dieser Hinweise können Sie dann den Satz korrigieren. Die Meldungen bleiben bis zum nächsten Öffnen oder Tragen erhalten.

Nach der Korrektur können Sie den gleichen Vorgang erneut aufrufen - es sind ja nur noch die zuerst fehlerhaften Sätze in der Zwischendatei. Bei Bedarf können Sie diesen Vorgang wiederholen, bis alle Sätze korrekt übernommen worden sind.

Wie die anderen Verarbeitungsfunktionen auch benutzt der Tragevorgang alle markierten Sätze, wenn mindestens ein Satz markiert ist - ansonsten alle durch die Suchbedingung gewählten. Das Tragen wird aufgerufen durch die Funktion

- Saetze tragen

Nach Eingabe des Zieldateinamens müssen Sie noch angeben, ob Sie die Prüfbedingungen testen wollen.

Zu diskutieren bleibt noch die Form der Prüfbedingungen. Diese stellen ein kleines ELAN-Programm dar, in dem einige spezielle Prozeduren zum Prüfen enthalten sind. Wenn Sie nicht ELAN programmieren können, sollte Sie diese Bemerkung nicht erschrecken: die Prüfbedingungen sind einfach genug.

Sie schreiben also die Prüfbedingungen jeweils untereinander. Eine mögliche Bedingung ist

```
wertemenge ("Feldname", "Wert1,Wert2,Wert3,Wert4");
```

Diese Bedingung gibt an, daß das Feld einen der angegebenen Werte haben muß. Die Werte werden untereinander durch Komma getrennt. Es gibt jedoch keine Möglichkeit, Werte mit Komma darzustellen, da das Komma immer als Trennung wirkt. Leerzeichen dürfen in den Werten vorkommen, sie müssen dann aber auch genau so im Feld stehen.

Wir könnten zum Beispiel eine Bedingung für unser Feld "m/w" wie folgt formulieren

```
wertemenge ("m/w", "m,w");
```

EUDAS würde sich dann beschweren, wenn das Feld leer wäre (irgendein Geschlecht muß die Person ja wohl haben). Wenn das Feld auch leer sein darf, geben Sie einfach zwei Kommata hintereinander oder ein Komma am Anfang an:

```
wertemenge ("m/w", ",m,w");
```

Eine andere Möglichkeit der Prüfbedingung besteht darin, eine Maske für ein Feld zu definieren. Diese Maske gibt an, daß an bestimmten Stellen des Feldes nur bestimmte Zeichen stehen dürfen. So könnte man zum Beispiel folgende Maske für ein Datumsfeld angeben:

```
feldmaske ("Datum", "99.99.99");
```

Die Neunen haben hier eine spezielle Bedeutung und und stehen für eine beliebige Ziffer. Es gibt noch einige weitere Zeichen, die eine reservierte Bedeutung haben, nämlich:

```
"9"   für jede Ziffer (wie schon erwähnt)
"X"   für jedes Zeichen
"A"   für jeden Großbuchstaben
"a"   für jeden Kleinbuchstaben
"*"   für eine Folge beliebiger Zeichen
```

Alle anderen Zeichen im Muster stehen für sich selbst. Eine Sonderstellung besitzt der Stern; er sollte sparsam verwendet werden, da seine Benutzung etwas aufwendiger ist. Der Stern kann auch für eine leere Zeichenfolge stehen. Als weiteres Beispiel könnte man definieren

```
feldmaske ("Name", "A*");
```

damit immer ein Name angegeben ist, der noch dazu mit einem Großbuchstaben beginnt.

Für Bedingungen, die sich nicht mit diesen beiden Prozeduren formulieren lassen, gibt es noch

```
pruefe ("Feldname", Bedingung);
```

Diese Prozedur erhält einen booleschen Wert als Parameter, der einen Vergleich darstellt. Ist dieser Parameter FALSE, wird eine entsprechende Fehlermeldung protokolliert. So könnte man folgende Bedingung angeben:

```
pruefe ("Alter", wert ("Alter") > 18.0);
```

Diese Bedingung würde sicherstellen, daß alle Personen in der Datei volljährig sind.

Da die Prüfbedingungen ein ELAN-Programm sind, können Sie natürlich sämtliche ELAN-Anweisungen verwenden.

Weiterhin haben Sie die Möglichkeit, Doppeleinträge zu verhindern. Dazu geben Sie mit Hilfe der Prozedur

```
eindeutige felder (n);
```

wieviele Felder vom ersten an eindeutig sein sollen. Ein zu tragender Satz, der mit irgendeinem Satz in diesen Feldern übereinstimmt, wird als fehlerhaft zurückgewiesen. In unserer Adressendatei könnte man

```
eindeutige felder (2);
```

angeben. Damit würde ein neuer Satz mit bereits vorhandenem Namen und Vornamen abgelehnt.

13
Kopieren und Bearbeiten

In diesem Kapitel werden Sie lernen, wie Sie EUDAS-Dateien kopieren und umstrukturieren können und wie Sie Änderungen auch automatisch durchführen können.

13.1 KOPIEREN

Manchmal taucht die Situation auf, daß Sie für einen bestimmten Zweck nur einen Ausschnitt einer EUDAS-Datei benötigen. Diesen Ausschnitt können Sie sich durch Kopieren herstellen.

Beim Kopieren wird ein Teil der Felder aller ausgewählten Sätze in eine andere Datei kopiert. Wie bei allen verarbeitenden Funktionen betrachtet das Kopieren alle markierten Sätze, wenn mindestens ein Satz markiert ist. Aufgerufen wird diese Funktion durch

- Saetze kopieren

Zum Kopieren wird die Information benötigt, welche Felder kopiert werden sollen. Diese Information wird durch ein **Kopiermuster** geliefert. Wenn Sie die Funktion aufrufen, wird Ihnen ein Kopiermuster zum Ändern angeboten.

Dieses Standard-Kopiermuster ist nun abhängig davon, ob die Zieldatei bereits existiert oder nicht. Die Zieldatei wird zuerst erfragt. Fall Sie nicht existiert, wird noch gefragt, ob sie eingerichtet werden soll.

Existiert die Datei noch nicht, so gibt das Standard-Kopiermuster an, daß alle Felder kopiert werden sollen. Die Zieldatei erhält die gleiche Feldstruktur wie die aktuelle Datei.

Existiert die Zieldatei dagegen bereits, werden nur die Felder im Kopiermuster angegeben, die auch in der Zieldatei vorkommen, so daß nur Felder der Zieldatei kopiert werden.

Damit Sie das Ganze auch selber ausprobieren können, müssen Sie natürlich erst wissen, wie ein Kopiermuster konkret aussieht. Ein Kopiermuster besteht aus Anweisungen der Form

```
"Feldname" K f ("Feldname");
```

Auf der linken Seite steht der Name des Feldes, in das kopiert werden soll; auf der rechten Seite steht das, was kopiert wird. Der angegebene Ausdruck bedeutet nichts anderes als 'Feldinhalt des Feldes "Feldname"'. Sie werden gleich sehen, daß dort auch noch andere Ausdrücke stehen können. Beachten Sie auch das Semikolon am Ende der Zeile. Es ist notwendig, um die einzelnen Anweisungen voneinander zu trennen.

Da Ihnen das Kopiermuster zum Ändern angeboten wird, können Sie es natürlich nach Ihren Wünschen modifizieren. Daher wollen wir Ihnen zuerst die generelle Wirkungsweise eines Kopiermusters erklären. Anschließend können wir dann die konkreten Aktionen für verschiedene Anwendungen besprechen.

Die Reihenfolge der "K"-Ausdrücke bestimmt die Reihenfolge der Feldnamen in der Zieldatei, die neu eingerichtet werden. Wenn also die Zieldatei noch nicht existierte, wird Sie mit den Feldern eingerichtet, die in den Ausdrücken auf der linken Seite angegeben sind.

Falls die Datei schon existiert, ein einzelnes angegebenes Feld jedoch noch nicht, wird dieses am Ende angefügt.

Der Ausdruck an der rechten Seite ist ein beliebiger ELAN-Ausdruck, der einen TEXT liefert (falls Sie ELAN nicht kennen, reichen Ihnen die hier dargestellten Ausdrücke). In Abschnitt 13.3 werden mögliche Ausdrücke besprochen.

Nun können wir einige Beispiele besprechen. Wenn Sie die Datei "Adressen" öffnen und in eine noch nicht existierende Datei kopieren wollen, wird Ihnen folgendes Standard-Kopiermuster angeboten:

```
"Name" K f ("Name");
"Vorname" K f ("Vorname");
"PLZ" K f ("PLZ");
"Ort" K f ("Ort");
"Strasse" K f ("Strasse");
"m/w" K f ("m/w");
```

Wenn Sie dieses Kopiermuster unverändert lassen, wird die Datei "Adressen" identisch kopiert.

Nehmen wir jetzt an, Sie wollen die Datei "Adressen" in eine andere Datei kopieren, die folgende Felder besitzt:

```
"Name"
"Vorname"
"Anrede"
```

Es würde Ihnen dann folgendes Kopiermuster angeboten:

```
"Name" K f ("Name");
"Vorname" K f ("Vorname");
"Anrede" K "";
```

Sie sehen also, daß nur die vorhandenen Felder angegeben sind. Außerdem wurde beim Feld "Telefon" der Ausdruck "" angegeben, der für den leeren Text steht. Dieses Feld ist ja in der Ausgangsdatei nicht enthalten. Diese letzte Zeile könnten Sie sich eigentlich sparen, da nicht kopierte Felder immer leer bleiben. Die Zeile wird jedoch angeboten, damit Sie sie gegebenenfalls ändern können.

Kommen wir zurück zum ersten Beispiel, in dem die Zieldatei noch nicht existierte. Sie können jetzt mit dem Kopiermuster bestimmen, wie die Zieldatei aussehen soll. Wenn Sie einzelne Zeilen des Kopiermusters löschen, werden die entsprechenden Felder auch nicht in die Zieldatei aufgenommen. Sie können auch die Reihenfolge der Zeilen ändern, und erzeugen damit auch ein andere Feldreihenfolge in der Zieldatei.

Gerade das letztere ist interessant, falls Sie eine Datei umstrukturieren wollen, damit zum Beispiel ein anderes Feld an erster Stelle erscheint (das erste Feld wird ja optimiert). Außerdem können Sie natürlich an jeder beliebigen Stelle einen neue Zeile einfügen, die dann einem neuen Feld entspricht.

Im zweiten Beispiel haben Sie nur noch die Möglichkeit, neue Felder am Ende anzufügen, da die Datei schon existiert. Sie können jedoch (wie auch im vorigen Beispiel) einen Ausdruck an der rechten Seite ändern. Hier kommt ins Spiel, daß das ganze Kopiermuster ein ELAN-Programm ist und entsprechend erweitert werden kann.

Um zum Beispiel statt "m" oder "w" die Inhalte "Herr" oder "Frau" zu erzeugen, könnten Sie das Kopiermuster wie folgt ändern:

```
"Name" K f ("Name");
"Vorname" K f ("Vorname");
"Anrede" K neue anrede.

neue anrede:
  IF f ("m/w") = "m" THEN
    "Herr"
  ELSE
    "Frau"
  END IF.
```

Wie in allen ELAN-Programmen können also auch in Kopiermustern zum Beispiel wertelieferende Refinements verwendet werden. In ähnlicher Weise könnten Sie unter Verwendung von ELAN-Anweisungen ein Feld in mehrere aufsplitten oder umgekehrt mehrere Felder zu einem zusammenfügen.

Ein Hinweis für Programmierer: da die Reihenfolge der "K"-Anweisungen wichtig ist, dürfen diese nie innerhalb einer IF-Anweisung stehen, sondern müssen für jeden bearbeiteten Satz in gleicher Reihenfolge ausgeführt werden.

Nach dieser ausführlichen Diskussion des Kopiermusters muß nun noch erklärt werden, wie der eigentliche Kopiervorgang abläuft. Zuerst wird natürlich das Kopiermuster vom ELAN-Compiler übersetzt. Wenn hier Fehler auftreten, wird der Vorgang abgebrochen und die Fehlermeldungen werden angezeigt.

Ist alles in Ordnung, werden die ausgewählten Sätze nach Angaben des Kopiermusters kopiert. Die Sätze werden dabei am Ende der Zieldatei angefügt. Bei der Bearbeitung wird jeweils die Nummer des bearbeiteten Satzes ausgegeben.

13.2 BEARBEITEN

Mit EUDAS können Sie die geöffnete Datei nicht nur satzweise von Hand ändern, sondern auch automatisch die ganze Datei. Dazu müssen Sie dem Rechner eine Vorschrift geben, nach der er handeln kann. Eine solche **Bearbeitungsvorschrift** stellt im Prinzip schon ein kleines Programm dar. Wenn Sie noch nie in ELAN programmiert haben, sollten Sie dieses Kapitel erst einmal überlesen.

Da auch ein Kopiermuster ein Programm war, ist es nicht erstaunlich, daß Bearbeitungsmuster ganz ähnlich aussehen. Eine typische Zeile sieht etwa so aus:

```
"Feldname" V "neuer Inhalt";
```

Diese Zeile bedeutet: Ersetze den Inhalt des Feldes "Feldname" durch den Text "neuer Inhalt". Anstelle des neuen Textes kann wieder ein beliebiger ELAN-Ausdruck stehen. Ein Beispiel, in dem ein Feld einen Stern angehängt bekommt, sieht dann so aus:

```
"Feldname" V f ("Feldname") + "*";
```

Beachten Sie, daß Sie den Ausdruck auf der rechten Seite eventuell in Klammern setzen müssen (obwohl der Operator "V" die niedrigste Priorität hat). Wenn Sie sich nicht sicher sind, können Sie den Ausdruck immer in Klammern einschließen.

Natürlich können Sie auch hier wie im Kopiermuster Refinements einsetzen. Nicht möglich sind jedoch Prozeduren und Pakete (da das Muster selbst eine Prozedur darstellt).

Nach dieser Diskussion des Bearbeitungsmusters wieder zurück zur Ausführung. Der Bearbeitungsvorgang wird durch

- nach Vorschrift aendern

aufgerufen. Dabei wird der Name des Bearbeitungsmusters erfragt. Dieses muß vorher in eine Textdatei geschrieben worden sein. Anschließend werden alle ausgewählten Sätze nach der Vorschrift bearbeitet. Dabei wird jeweils die aktuelle Satznummer ausgegeben.

13.3 ELAN-AUSDRÜCKE

In diesem Abschnitt soll zum einen erklärt werden, welche spezifischen Ausdrücke EUDAS zur Verfügung stellt. Zum anderen sollen aber auch die wichtigsten Ausdrücke zusammengefaßt werden, die das EUMEL-System definiert, die aber in EUDAS sehr häufig angewendet werden.

Sie haben bereits einen Ausdruck zur Abfrage von Feldinhalten der aktuellen Datei kennengelernt:

```
f ("Feldname")
```

Diese Prozedur liefert den Inhalt des angegebenen Feldes bezogen auf den aktuellen Satz. Der Feldname muß natürlich existieren, ansonsten wird ein Abbruch erzeugt. Ähnlich liefert die Prozedur 'wert' den Wert eines Feldinhaltes als REAL-Zahl, falls Sie damit rechnen wollen.

Ein TEXT-Ausdruck kann auch als Verkettung mehrere TEXT-Ausdrücke mit '+' entstehen. Im vorigen Abschnitt sahen Sie ein Beispiel dafür. Umgekehrt liefert

```
subtext (f ("Feldname"), von, bis)
```

den Ausschnitt eines Feldinhalts von der Position 'von' bis einschließlich der Position 'bis' (diese beiden Angaben sind einfache INT-Zahlen).

Um komplizierte Möglichkeiten zu realisieren, können Sie in einem Refinement auch IF-Anweisungen einsetzen. Dazu definieren Sie am Ende des Musters etwa solch ein Refinement:

```
name fuer den ausdruck:
  IF Bedingung THEN
    "Text 1"
  ELSE
    "Text 2"
  END IF.
```

Der 'name fuer den ausdruck' kann dann weiter oben anstelle eines Ausdrucks eingesetzt werden. Die Bedingung wird in der regel ein Vergleich von zwei Texten sein (dies können wieder beliebige TEXT-Ausdrücke sein). Auch im THEN- und im ELSE-Teil können beliebige TEXT-Ausdrücke angegeben werden und nicht nur einfache Texte wie hier.

Denken Sie jedoch bei IF-Anweisungen (und auch SELECT-Anweisungen) daran, daß der Ausdruck immer einen Wert liefern muß, also alle Möglichkeiten abgedeckt sein müssen.

Weitere Möglichkeiten, wertelieferende Ausdrücke zu konstruieren, finden Sie in der Literatur zur Programmiersprache ELAN.

14
Programmierung von Druckmustern

In den nächsten Kapiteln wollen wir uns mit der Möglichkeit beschäftigen, den Leistungsumfang des Druckmusters durch ELAN-Anweisungen zu erweitern. Diese Erweiterung geschieht in der Form von Abkürzungen und zusätzlichen Anweisungen.

Um diese Möglichkeiten richtig zu nutzen, sollten Sie eine gewisse Erfahrung im Erstellen von ELAN-Programmen besitzen. Die vorgestellten Beispiele decken jedoch einen weiten Bereich von Anwendungen ab, so daß Sie auch ohne Programmierkenntnisse Nutzen aus diesen Kapiteln ziehen können.

14.1 ABKÜRZUNGEN

In den vorigen Kapiteln haben Sie erfahren, daß man Feldmuster von ganz bestimmter Länge definieren kann, deren Inhalt in genau dieser Länge eingesetzt und bei Bedarf abgeschnitten wird. Bei der Angabe dieser Länge spielt jedoch die Länge des Feldnamens eine ganz entscheidende Rolle. Das kürzeste Feldmuster fester Länge, das Sie definieren können, ist nämlich zwei Zeichen länger als der Feldname (ein Musterzeichen vorher und eins nachher).

Hätte das Feld "PLZ" den Namen "Postleitzahl" bekommen, so müßte ein solches Feldmuster mindestens eine Länge von 14 Zeichen haben. Damit Sie mit diesem Feldnamen auch ein Feldmuster der Länge 4 bekommen können (Postleitzahlen haben in den seltensten Fällen mehr als 4 Stellen), haben Sie die Möglichkeit, den Namen "Postleitzahl" für die Verwendung im Druckmuster geeignet abzukürzen.

Abkürzungen haben jedoch noch eine viel weitreichendere Bedeutung. Mit ihnen ist es möglich, nicht nur die Feldinhalte einer EUDAS-Datei einzusetzen, sondern auch jeden anderen Text, den Sie mit einem ELAN-Programm erzeugen können.

Die einfachsten zusätzlichen Daten, die Sie verwenden können, sind z.B. Datum und Uhrzeit. Für weitergehende Zwecke können Sie die Inhalte der EUDAS-Datei auch für Berechnungen verwenden und damit so umfangreiche Probleme wie das

Schreiben von Rechnungen oder statistische Auswertungen unter Verwendung eines Druckmusters lösen.

Abkürzungen werden in einem speziellen Abkürzungsteil am Ende eines Abschnittes angegeben. Der Abkürzungsteil wird durch die Anweisung

```
% ABKUERZUNGEN
```

eingeleitet. Eine Abkürzungsdefinition hat eine ähnliche Form wie ein Refinement (Falls Sie nicht wissen, was das ist, vergessen Sie es). Zu Beginn steht der Name der Abkürzung in Form eines Feldmusters, beginnend in der ersten Spalte. Danach folgt, durch Leerzeichen getrennt, ein Doppelpunkt in der gleichen Zeile. Daran schließt sich ein beliebiger ELAN-Ausdruck an, der sich in freiem Format über beliebig viele Zeilen erstrecken kann und mit einem Punkt abgeschlossen werden muß. Dieser ELAN-Ausdruck muß ein TEXT-Objekt liefern.

Für die Abfrage von Inhalten aus einer EUDAS-Datei ist der Ausdruck

```
f ("Feldname")
```

vordefiniert. Die Abkürzung des Feldes "Postleitzahl" würde also als Ausschnitt folgendermaßen aussehen:

```
% ABKUERZUNGEN
&p :   f ("Postleitzahl") .
```

Mit dieser Definition kann man im Muster so verfahren, als ob das Feld "Postleitzahl" auch "p" hieße. Diese einfachste Form der Abkürzung können Sie natürlich variieren, indem Sie für "p" und "Postleitzahl" Ihre eigenen Namen einsetzen.

Um die Verwendung von Abkürzungen zu demonstrieren, wollen wir folgendes Druckmuster betrachten:

```
% VORSPANN
Adressenliste als Beispiel für Abkürzungen
Stand: &Datum
-----------------------------------------
% ABKUERZUNGEN
&Datum : date .
```

```
% NIEDERHOLUNG
&&l :   &Vorname %Name
        &Strasse
  &&p& &Ort
-----------------------------------------
% ABKUERZUNGEN
&l : lfd nr .
&p : f ("PLZ") .

% NACHSPANN
&l Adressen gedruckt.
```

Dieses Beispiel enthält eine ganze Reihe interessanter Details. Als erstes sollten Sie registrieren, daß auch im Vorspann oder Nachspann Feldmuster verwendet werden können. Soll in diesem Fall ein Feldinhalt aus der EUDAS-Datei eingesetzt werden, so werden beim Vorspann die Inhalte des ersten und beim Nachspann die Inhalte des letzten durch Suchmuster ausgewählten Satzes verwendet. Daher kann auch jeder Abschnitt einen Abkürzungsteil haben. Abkürzungen gelten jedoch für alle Abschnitte (s. "&l"); die Aufteilung in mehrere Abkürzungsteile fördert im wesentlichen die Übersichtlichkeit.

Versuchen Sie, an diesem Beispiel die wichtigsten Unterschiede zwischen dem **Musterteil** und dem **Abkürzungsteil** eines Abschnittes zu verstehen. Das Format des Musterteiles soll in die Ausgabe übernommen werden; daher ist dort die Stellung jedes Wortes wichtig. Im Abkürzungsteil definieren Sie Abkürzungen ohne bestimmtes Format - mit der einzigen Ausnahme, daß eine Abkürzungsdefinition mit einem "&" in der ersten Spalte anfangen muß. Wie Sie sehen, dürfen dort Leerzeilen zur besseren Lesbarkeit eingefügt werden.

Bevor wir die einzelnen Abkürzungen des Druckmusters besprechen, sollten Sie das Beispiel ausprobieren.

Beachten Sie dabei, daß das Druckmuster in ein ELAN-Programm umgeformt werden muß, da ELAN-Ausdrücke in ihm vorkommen. Das automatisch erzeugte ELAN-Programm wird dann vom ELAN-Compiler übersetzt und ausgeführt. Fehler in den ELAN-Ausdrücken im Abkürzungsteil können erst vom ELAN-Compiler entdeckt werden. Dieser kennt jedoch das Druckmuster nicht und meldet die Fehler anhand des generierten Programms. Sie müssen in einem solchen Fall aufpassen, daß Sie die Fehlerquelle an der richtigen Stelle im Druckmuster lokalisieren (Hilfestellungen dazu sind im Kapitel über die Übersetzung von Druckmustern zu finden).

Sie sollten bei unserem Beispiel folgende Ausgabe erhalten:

```
Adressenliste als Beispiel für Abkürzungen
Stand:  28.12.84
----------------------------------------
  1 :   Herbert Wegner
        Krämergasse 12
  5000  Köln
----------------------------------------
  2 :   Helga Sandmann
        Willicher Weg 109
  5300  Bonn 1
----------------------------------------
  3 :   Albert Katani
        Lindenstr. 3
  5210  Troisdorf
----------------------------------------
  4 :   Peter Ulmen
        Mozartstraße 17
     5  Köln 60
----------------------------------------
  5 :   Karin Regmann
        Grengelweg 44
  5000  Köln 90
----------------------------------------
  6 :   Hubert Arken
        Talweg 12
  5200  Siegburg
----------------------------------------
  7 :   Anna-Maria Simmern
        Platanenweg 67
     5  Köln 3
----------------------------------------
  8 :   Angelika Kaufmann-Drescher
        Hauptstr. 123
    53  Bonn 2
----------------------------------------
  9 :   Harald Fuhrmann
        Glockengasse 44
  5000  Köln 1
----------------------------------------
 10 :   Friedrich Seefeld
        Kabelgasse
  5000  Köln-Ehrenfeld
----------------------------------------
10 Adressen gedruckt.
```

Nun zu den Abkürzungen im einzelnen. Das Feld "PLZ" muß abgekürzt werden, damit es rechtsbündig vor den Ort gedruckt werden kann. Die Abkürzung "p" benutzt die im vorigen Kapitel beschriebene Form zur Abfrage des Feldinhaltes.

"Datum" wird als Abkürzung für das aktuelle Datum definiert, ein häufig benötigter Fall. 'date' ist der ELAN-Ausdruck, der das Datum liefert. (Bemerkung für ELAN-Programmierer: der Name der Abkürzung gehorcht nicht der ELAN-Syntax für Bezeichner).

Eine für Tabellen sinnvolle Funktion wird bei der Definition von "l" verwendet. Der von EUDAS definierte Ausdruck 'lfd nr' liefert die laufende Nummer des gerade gedruckten Satzes als Text. Dabei ist zu beachten, daß die laufende Nummer nicht mit der Satznummer übereinstimmt, sondern nur während des Druckvorganges von 1 an bei jedem gedruckten Satz hochgezählt wird. Diese Funktion dient dazu, die Sätze in der Liste durchzunumerieren.

Die laufende Nummer soll in der Liste rechtsbündig mit Doppelpunkt vor dem Namen stehen. Dazu wird das Feldmuster "&&l" benutzt, eine Form, die eigentlich keinen Sinn hat (die Kombination 'variable Länge' und 'rechtsbündig' gibt es nicht). Um ein möglichst kurzes Feldmuster schreiben zu können, wird in diesem Fall jedoch feste Länge unterstellt (auch ohne folgendes "&"). Damit hat das kürzeste Feldmuster fester Länge drei Zeichen sowohl im rechtsbündigen ("&l&") wie auch im linksbündigen Fall ("&&l").

Die Verwendung der Abkürzung "l" im Nachspann kann als erstes Beispiel für eine Auswertungsfunktion gelten. Da für den Nachspann die Daten des letzten Satzes verwendet werden, erscheint hier die laufende Nummer des letzten Satzes und somit die Anzahl der Sätze, die gedruckt wurden. Das kann dazu benutzt werden, die Sätze zu zählen, die eine bestimmte Suchbedingung erfüllen. Folgendes Druckmuster zählt die Anzahl der Frauen oder Männer in der Datei zu zählen:

```
% NACHSPANN
&l Personen mit dem Geschlecht "%<m/w>" vorhanden.
% ABKUERZUNGEN
&l : lfd nr .
```

Wenn Sie vor dem Drucken jetzt die Suchbedingung "m" für das Feld "m/w" einstellen, werden alle Männer ausgewählt. Das Drucken besteht in diesem Fall nur aus dem Hochzählen der

laufenden Nummer für jeden Mann. Im Nachspann kann das Ergebnis dann ausgegeben werden; zugleich soll der aktuelle Wert des Feldes "m/w" gedruckt werden, damit das Druckmuster auch für das Zählen der Frauen verwendet werden kann.

Die beiden möglichen Ausgaben würden dann so aussehen:

```
6 Personen mit dem Geschlecht "m" vorhanden.

4 Personen mit dem Geschlecht "w" vorhanden.
```

Wir können die Erkenntnisse dieses Abschnittes wie folgt zusammenfassen:

- Feldmuster können auch im Vorspann und Nachspann verwendet werden. Im Vorspann werden die Daten des ersten, im Nachspann die Daten des letzten ausgewählten Satzes verwendet.
- Der Musterteil eines Abschnittes definiert ein Format; der Abkürzungsteil ist formatfrei.
- 'lfd nr' dient zum Durchnumerieren aller gedruckten Sätze.
- Ein rechtsbündiges Feldmuster hat immer auch feste Länge.

14.2 KOMPLEXE ABKÜRZUNGEN

Mit Hilfe von Abkürzungen können wir jetzt auch bessere Musterbriefe schreiben. Ein Problem, das bereits angesprochen wurde, besteht darin, daß in der Anrede je nach Geschlecht "Herr" oder "Frau" stehen soll. Um dieses Problem zu lösen, wird der Inhalt des Feldes "m/w" benötigt.

Da in einer Abkürzung jede ELAN-Anweisung erlaubt ist, die einen Text liefert, können natürlich auch **IF-Anweisungen** verwendet werden. Mit diesen Informationen können wir jetzt die Abkürzung "Anrede" definieren:

```
% ABKUERZUNGEN
&Anrede :
  IF f ("m/w") = "w" THEN
    "Frau"
  ELSE
    "Herr"
  END IF .
```

Für Nicht-Programmierer: Die IF-Anweisung besteht aus einer Abfrage und zwei Alternativen. Die Abfrage steht zwischen dem IF und dem THEN und besteht in der Regel aus einer Abfrage, ob zwei Dinge gleich oder ungleich (<>), größer oder kleiner sind. Außerdem können mehrere Abfragen mit AND (und) und OR (oder) kombiniert werden. Näheres dazu im Benutzerhandbuch, Teil 6.

Die Alternative hinter dem THEN wird ausgewählt, wenn die Abfrage zutrifft. An dieser Stelle sind wieder beliebige Ausdrücke erlaubt, die einen Text liefern, einschließlich erneuter IF-Anweisungen (Schachtelung). Die Alternative zwischen ELSE und END IF wird ausgewählt, wenn die Abfrage nicht zutrifft.

Bisher wurden nur ELAN-Funktionen als Textlieferanten betrachtet ('date', 'lfd nr', 'f'). In unserem Fall werden aber **Textkonstanten** in den Alternativen der IF-Anweisung benötigt. Textkonstanten werden in ELAN in Anführungsstriche eingeschlossen, die aber nicht zum Text gehören. Innerhalb einer Textkonstanten werden Leerzeichen wie alle anderen Zeichen angesehen (erscheinen also auch nachher in der Ausgabe).

Bei solchen Abkürzungen, die längere Anweisungen umfassen, sollten Sie das freie Format ausnutzen und eine möglichst übersichtliche Darstellung wählen. Wie Sie sehen, muß nur der Doppelpunkt noch in der ersten Zeile stehen, der Rest kann sich beliebig auf die folgenden Zeilen erstrecken.

Ein typischer Einsatz einer IF-Anweisung für die Anrede sieht so aus:

```
% WIEDERHOLUNG

Sehr geehrte&Anrede %Name !

...
% ABKUERZUNGEN
&Anrede :
  IF f ("m/w") = "m" THEN
    "r Herr"
  ELSE
    " Frau"
  END IF .
```

Sie sollten jetzt diese Konstruktion in einen Musterbrief einfügen können. Probieren Sie ihn dann als Beispiel aus !

Durch Verwendung von Abkürzungen ist es auch möglich, rechtsbündige Felder mit einer Länge von weniger als 3 Zeichen zu simulieren. Dies geschieht mit Hilfe der Textoperationen von ELAN. Ohne ELAN-Vorkenntnisse können Sie dieses Beispiel überlesen. In unserer Liste im obigen Beispiel sind die laufenden Nummern höchstens zweistellig und sollten deshalb auch nur zwei Stellen belegen. Dies würde folgende Abkürzung ermöglichen:

```
% ABKUERZUNGEN
&l :  text (lfd nr als zahl, 2)
lfd nr als zahl : int (lfd nr) .
```

Die Prozedur 'text' wird dazu benutzt, eine Zahl rechtsbündig auf zwei Stellen zu formatieren (s. EUMEL-Benutzerhandbuch). Da die Abkürzung immer eine Länge von zwei Zeichen hat, kann sie auch in einem Feldmuster variabler Länge eingesetzt werden. Die Attribute 'feste Länge' und 'rechtsbündig' werden in diesem Fall also nicht durch das Feldmuster, sondern durch die Abkürzung selbst erzeugt.

Um die Prozedur 'text' anwenden zu können, muß die laufende Nummer als Zahl (sprich: INT-Objekt) vorliegen. Diese Umwandlung wird mit der Prozedur 'int' vorgenommen, die einen Text in eine Zahl umwandelt. Obwohl man 'int (lfd nr)' direkt in den Aufruf von 'text' hätte schreiben können, wird hier als Demonstration dafür ein Refinement verwendet.

Refinements können in einem Abkürzungsteil neben Abkürzungen stehen und von allen Abkürzungen benutzt werden. Sie werden ähnlich geschrieben wie Abkürzungen, nur ihr Name muß in Kleinbuchstaben geschrieben werden, dafür muß er nicht in der ersten Spalte anfangen und kann Leerzeichen enthalten. Bei komplizierteren Ausdrücken sollten Refinements zur besseren Lesbarkeit eingesetzt werden.

Sie können die IF-Anweisung auch mit beliebig vielen ELIF-Teilen versehen. Achten Sie jedoch darauf, daß die IF-Anweisung immer irgendeinen Wert liefern muß. Sie dürfen also den ELSE-Teil nicht weglassen. Statt einer IF-Anweisung können Sie natürlich auch eine SELECT-Anweisung verwenden. Es stehen Ihnen im Prinzip alle wertelieferndn Anweisungen von ELAN zur Verfügung.

Die Programmiersprache ELAN bietet Ihnen noch weit mehr Möglichkeiten, als hier beschrieben werden können. So können Sie sich eigene Prozeduren definieren und diese dann in Ab-

kürzungen verwenden. Wenn Sie ELAN genauer kennenlernen wollen, sollten Sie sich mit dem Benutzerhandbuch und weiterer Literatur beschäftigen. Sie werden anschließend auch die Fähigkeiten des Druckmusters besser nutzen können.

14.3 BEDINGTE MUSTERTEILE

Wenn größere Teile des Druckmusters in Abhängigkeit von bestimmten Daten unterschiedlich ausfallen sollen, werden die dazu benötigten Abkürzungen sehr umfangreich. Für solche Fälle kann man IF-Anweisungen auch im Musterteil eines Abschnitts verwenden. In diesem Fall werden die Alternativen der IF-Anweisung durch Musterzeilen dargestellt.

Im Musterteil müssen jedoch die Zeilen, die Teil der IF-Anweisung sind, von den Musterzeilen unterschieden werden. Deshalb werden die Anweisungszeilen durch ein "%"-Zeichen in der ersten **und** zweiten Spalte gekennzeichnet. Das zweite "%"-Zeichen dient zur Unterscheidung von Anweisungen an den Druckgenerator, die nicht an den ELAN-Compiler übergeben werden sollen.

Mit einer IF-Anweisung im Musterteil kann man das Anredeproblem auch folgendermaßen lösen:

```
% WIEDERHOLUNG
%% IF f ("m/w") = "w" THEN
Sehr geehrte Frau &Name !
%% ELSE
Sehr geehrter Herr &Name !
%% END IF;
```

Beachten Sie den Unterschied, daß die IF-Anweisung hier mit einem Semikolon abgeschlossen werden muß - in Abkürzungen mußte ja ein Punkt danach folgen. Außerdem darf hier der ELSE-Teil (die zweite Alternative) fehlen, während in einer Abkürzung in jeder Alternative etwas stehen muß (zumindest der leere Text "").

Falls sich der IF-THEN-Teil über mehr als eine Zeile erstrecken soll, muß jede dieser Zeilen mit "%%" beginnen, da die Folgezeilen sonst als Musterzeilen gedruckt würden. Benutzen Sie in einem solchen Fall jedoch besser ein Refinement, das Sie im Abkürzungsteil definieren müssen.

Sie können im Musterteil auch andere ELAN-Anweisungen verwenden. Der Unterschied zu Abkürzungen liegt darin, daß die Musterzeilen nicht als Werte angesehen werden, die die

Anweisung liefern muß, sondern als Anweisungen, die dort aufgeführten Musterzeilen einzusetzen und zu drucken. Daher kann im Musterteil auch eine FOR-Schleife sinnvoll sein, wenn in Abhängigkeit eines Wertes eine bestimmte Anzahl von Zeilen gedruckt werden soll.

14.4 ÜBERSETZUNG

Wenn Sie bis jetzt auch als ELAN-Programmierer immer noch nicht ganz durchblicken, wie Sie welche ELAN-Anweisungen verwenden können, dann ist das noch kein Anlaß zur Sorge. Es ist kaum möglich, die genauen Auswirkungen beliebiger Anweisungen zu beschreiben, ohne den Übersetzungsprozeß zu schildern, der diese Anweisungen zu einem ELAN-Programm zusammenbindet. Daher soll diese Übersetzung jetzt genauer erklärt werden.

Alle Zeilen eines Abkürzungsteils werden direkt in das Programm übernommen, wobei der Name einer Abkürzung durch einen beliebig gewählten Refinementnamen ersetzt wird ("abk" + eine laufende Nummer). Alle Abkürzungen und Refinements werden als globale Refinements definiert, also außerhalb von Prozeduren. Dadurch wird erreicht, daß sie an jeder Stelle verwendet werden können.

Damit eine Abkürzung richtig als Refinement übersetzt wird, muß sie ein TEXT-Objekt als Wert liefern. Die anderen Refinements sind beliebig, da Sie nur in selbstdefinierten Anweisungen verwendet werden. Die Refinements der Abkürzungen werden in einer Zuweisung an eine TEXT-Variable verwendet, damit der Druckgenerator auf den entsprechenden Wert zugreifen kann.

Jeder Abschnitt wird dagegen als eine Prozedur übersetzt. Jede Folge von Musterzeilen wird in eine Anweisung übersetzt, diese Musterzeilen einzusetzen und zu drucken. Jede "%%"-Anweisung wird einfach unverändert dazwischen geschrieben. Die Vorspann-Prozedur wird einmal zu Anfang aufgerufen, die Prozedur für den Wiederholungsteil einmal für jeden ausgewählten Satz und die Nachspann-Prozedur einmal am Schluß.

Findet sich in dem erzeugten ELAN-Programm ein Fehler, der durch den Druckgenerator nicht erkannt werden konnte (z.B. eine Abkürzung liefert keinen Wert), so muß der ELAN-Compiler diesen Fehler erkennen. Anschließend zeigt er das

erzeugte Programm zusammen mit seinen Fehlermeldungen im Paralleleditor. Sie müssen nun die Fehlermeldung lokalisieren und anhand der eben gegebenen Hinweise in das ursprüngliche Druckmuster zurückübersetzen, damit Sie dort den Fehler korrigieren können.

Nun müßten Sie genug Informationen haben, um beliebige ELAN-Anweisungen in das Druckmuster einfügen zu können. Als Beispiel wollen wir versuchen, alle Männer und Frauen in der Adressendatei zu zählen, ohne ein Suchmuster einstellen zu müssen und ohne den Druckvorgang zweimal ablaufen zu lassen (wie dies bei dem obigen Beispiel der Fall war).

Ein erster Versuch könnte so aussehen:

```
% VORSPANN
%% INT VAR maenner, frauen;
%% maenner := 0;
%% frauen := 0;
% WIEDERHOLUNG
%% IF f ("m/w") = "m" THEN
%%   maenner INCR 1
%% ELSE
%%   frauen INCR 1
%% END IF
% NACHSPANN
&maenner Männer und %frauen Frauen vorhanden .
```

Aber Vorsicht ! In diesem Beispiel sind mehrere Fehler eingebaut. Finden Sie sie !

Der erste Fehler befindet sich im Nachspann. Hier wird versucht, die Namen der beiden Variablen 'maenner' und 'frauen' direkt in einem Feldmuster zu verwenden. Diese beiden Namen sind dem Druckgenerator nicht bekannt, sondern nur dem ELAN-Compiler. Um die Werte der beiden Variablen einsetzen zu können, müssen Sie also zwei geeignete Abkürzungen definieren.

Der zweite Fehler ist schwieriger zu finden. Wie oben gesagt, wird jeder Abschnitt in eine Prozedur übersetzt. Die in einem Abschnitt definierten Variablen können also nur in diesem Abschnitt verwendet werden (sie sind lokal) und auch nicht im Abkürzungsteil, da dieser wieder global vereinbart wird. Die beiden im Vorspann definierten Variablen stehen also im Wiederholungsteil und im Nachspann nicht zur Verfügung.

Für diesen Fall gibt es die Möglichkeit, ELAN-Anweisungen vor allen Abschnitten im sogenannten Initialisierungsteil zu definieren. Diese Anweisungen sind dann ebenfalls global. Das richtige Druckmuster würde also so aussehen:

```
%% INT VAR maenner, frauen;
% VORSPANN
%% maenner := 0;
%% frauen := 0;
% WIEDERHOLUNG
%% IF f ("m/w") = "m" THEN
%%   maenner INCR 1
%% ELSE
%%   frauen INCR 1
%% END IF
% NACHSPANN
&m Männer und %f Frauen vorhanden .
% ABKUERZUNGEN
&m : text (maenner)
&f : text (frauen) .
```

Natürlich könnten Sie die Initialisierung der beiden Variablen auch noch aus dem Vorspann herausnehmen. Denken Sie daran, daß Sie aus INT-Variablen erst einen Text machen müssen, ehe Sie sie in eine Musterzeile einsetzen können. Beachten Sie Schreibweise der Variablen: in ELAN können die Umlaute nicht in Bezeichnern verwendet werden, daher muß die Variable mit "ae" geschrieben werden. Im Mustertext und in Abkürzungs- und Feldnamen können die Umlaute jedoch frei verwendet werden.

15
Drucken für Fortgeschrittene

Dieses Kapitel beschreibt einige Fähigkeiten, die Sie erst mit einiger Erfahrung ausschöpfen können. Zugleich wird die Interpretation der Abschnitte und Feldmuster noch einmal zusammengefaßt, weil erst dann alle notwendigen Informationen vorliegen. Diese Beschreibung soll dem Fortgeschrittenen ermöglichen, im Zweifelsfall die genaue Wirkung eines Druckmusters feststellen zu können.

15.1 GRUPPEN

Der Druckgenerator bietet die Möglichkeit, Vorspann und Nachspann nicht nur am Anfang und am Ende, sondern auch an bestimmten Stellen zwischen Sätzen zu drucken. Diese Stellen sind dadurch bestimmt, daß ein bestimmtes Merkmal (z.B. ein Feldinhalt) seinen Wert ändert. Ein solches Merkmal wird im Druckmuster **Gruppe** genannt.

Ein Beispiel für die Verwendung von Gruppen ist eine Schülerdatei, die nach Klassen geordnet ist. Definiert man das Feld "Klasse" als Gruppe, so wird jeweils am Ende einer Klasse ein Nachspann und am Beginn einer Klasse ein Vorspann gedruckt.

Dieses Verfahren ist eine Erweiterung der bisher beschriebenen Methode, indem eine Datei quasi in mehrere Dateien untergliedert wird, die jedoch in einem Arbeitsgang gedruckt werden können. Voraussetzung dafür ist jedoch, daß die Datei nach dem Gruppenmerkmal geordnet ist - der Druckgenerator sammelt nicht erst alle Schüler einer Klasse aus der Datei, sondern erwartet sie hintereinander.

Eine Gruppe wird im Initialisierungsteil des Druckmusters (also vor allen Abschnitten) definiert. Notwendige Daten sind eine Nummer zur Identifizierung und das Merkmal. Die Nummer sollte am sinnvollsten von 1 an vergeben werden; die möglichen Werte sind nach oben hin beschränkt. Das Merkmal ist ein beliebiger ELAN-Ausdruck, der einen Text liefert. Sinnvollerweise wird er den Inhalt eines Feldes enthalten.

Gruppendefinitionen müssen nach allen ELAN-Anweisungen im Initialisierungsteil folgen, und zwar, weil die Gruppendefinitionen alle in einer Prozedur zusammengefaßt werden, die bei jedem neuen Satz auf Gruppenwechsel testet.

Unter der Annahme, daß die oben erwähnte Schülerdatei ein Feld "Klasse" besitzt, würde die Gruppe wie folgt definiert:

```
% GRUPPE  1  f ("Klasse")
```

Nach der Anweisung "GRUPPE" folgt die Gruppennummer und dann ein ELAN-Ausdruck. Die ganze Definition muß in einer Zeile stehen; reicht der Platz nicht aus, müssen Sie in einem Abkürzungsteil ein Refinement definieren.

Das komplette Druckmuster für die Klassenliste könnte folgendes Aussehen haben, wenn außer "Klasse" auch noch die Felder "Name" und "Vorname" vorhanden sind:

```
% GRUPPE  1  f ("Klasse")
% VORSPANN
Klassenliste für Klasse &Klasse
----------------------------
% WIEDERHOLUNG
&Vorname %Name
% NACHSPANN
#page#
```

Wenn eine Gruppe definiert ist, werden im Nachspann immer die Feldinhalte des letzten Satzes vor dem Gruppenwechsel gedruckt, im Vorspann die Inhalte des ersten Satzes nach dem Wechsel. Daher kann hier im Vorspann die Klasse gedruckt werden, da sie sich erst ändert, wenn schon wieder der nächste Vorspann gedruckt wird.

Wie die Identifikation über eine Gruppennummer vermuten läßt, können Sie mehrere Gruppen definieren. Nachspann und Vorspann werden jeweils gedruckt, wenn sich das Merkmal irgendeiner Gruppe ändert. Ob eine bestimmte Gruppe gewechselt hat, kann mit der Abfrage

```
BOOL PROC gruppenwechsel (INT CONST gruppennummer)
```

in einer IF-Anweisung ermittelt werden. Vor dem ersten und nach dem letzten Satz wechseln automatisch alle Gruppen.

Die ganze Datei bildet eine Quasi-Gruppe mit der Nummer 0. Sie ist immer definiert und wechselt nur vor dem ersten und nach dem letzten Satz. Sie ist es, die bewirkt, daß Vorspann und Nachspann in ihrer normalen Weise gedruckt werden.

Um einige der Möglichkeiten zu illustrieren, die durch Gruppen geschaffen werden, wollen wir als Beispiel eine Anwendung betrachten, die neue Wege für die Benutzung von EUDAS aufzeigt.

Aus einer Datei, in der für jede Bestellung der Kunde, der Artikel, die bestellte Menge und der Einzelpreis des Artikels eingetragen werden, sollen anschließend Rechnungen gedruckt werden. Die Datei soll folgende Felder haben:

```
"Kundennummer"
"Artikelnummer"
"Einzelpreis"
"Menge"
```

Als Voraussetzung müssen die Bestellungen in der Datei jeweils nach Kunden geordnet vorliegen. Die Kundennummer wird als Gruppe definiert, so daß die Bestellungen eines Kunden zu einer Rechnung zusammengefaßt werden können. Das Druckmuster rechnet dann die einzelnen Preise zusammen und gibt eine Endsumme aus.

Damit in der Rechnung Name und Adresse des Kunden auftauchen können, wird zu der Bestellungsdatei die Kundendatei gekoppelt, die folgende Felder haben soll:

```
"Kundennummer"
"Name"
"Vorname"
"Strasse"
"PLZ"
"Ort"
```

Stellen Sie sich zum Ausprobieren des folgenden Druckmusters gegebenenfalls eigene Daten zusammen. Hier nun das Druckmuster:

```
%% REAL VAR gesamtpreis, summe;
% GRUPPE  1  f ("Kundennummer")
% VORSPANN
%% summe := 0.0;
                                        Fa. Kraus & Sohn
                                        Schotterstr. 10

                                        5000 Köln 1
&Vorname %Name
&Strasse

&PLZ &Ort                               &Datum

                R E C H N U N G
                ===============

Menge  Artikelnr.        Einzelpreis  Gesamtpreis
-----------------------------------------------
% ABKUERZUNGEN
&Datum : date .

% WIEDERHOLUNG
%% gesamtpreis := round
%%   (wert ("Einzelpreis") * wert ("Menge"), 2);
%% summe INCR gesamtpreis;
&Menge &Artikelnummer     &&&&epr&&    &&&&gpr&&
% ABKUERZUNGEN
&epr :   f ("Einzelpreis") .
&gpr :   text (gesamtpreis, 9, 2) .

% NACHSPANN
-----------------------------------------------
                          Summe:         &&&&sum&&
                          + 14% MWSt.    &&&Mwst&&
                                         =========
                          Endbetrag      &&&&end&&
#page#
% ABKUERZUNGEN
&sum  : text (summe, 9, 2) .
&Mwst : text (mwst, 9, 2) .
&end  : text (summe + mwst, 9, 2) .
mwst  : round (summe * 0.14, 2) .
```

Im Initialisierungsteil dieses Druckmusters wird die Kundennummer als Gruppe definiert. Dies hat zur Folge, daß für jeden neuen Kunden eine neue Rechnung begonnen wird, nachdem vorher im Nachspann die Rechnungssumme des vorherigen Kunden berechnet und ausgedruckt wurde. Vor dieser Gruppendefinition sind 'gesamtpreis' und 'summe' definiert, die später als globale Variablen zur Verfügung stehen sollen. Diese Zeile darf nicht nach der Gruppendefinition stehen.

Im Vorspann wird der Kopf der Rechnung angegeben. Dieser enthält neben den Daten des Kunden (aus der gekoppelten Kundendatei) noch das Datum. Die Kundennummer wird nur zum Koppeln und als Gruppenmerkmal benötigt, erscheint also nicht auf der Rechnung.

Es fällt auf, daß im Firmennamen ein "&"-Zeichen auftaucht, das doch für die Markierung von Feldmustern reserviert ist. Die beiden Musterzeichen können jedoch im normalen Text auftauchen, wenn ihnen direkt ein Leerzeichen folgt. In diesem Fall werden Sie nicht als Beginn eines Feldmusters interpretiert, sondern unverändert gedruckt. Der gleiche Fall taucht noch mit "%" im Nachspann auf.

Im Wiederholungsteil wird zunächst aus dem Einzelpreis und der Menge des jeweiligen Artikels der Gesamtpreis für diesen Artikel berechnet. Für die Abfrage der Feldinhalte wird die Funktion 'wert' verwendet, die wie 'f' funktioniert, jedoch gleich einen REAL-Wert liefert.

Damit kaufmännisch richtig gerechnet wird, wird der Gesamtpreis auf 2 Nachkommastellen gerundet und erst dann aufsummiert. Würde der Gesamtpreis nur zum Einsetzen gerundet, könnten bei der anschließenden Addition der nicht gerundeten Werte eine falsche Gesamtsumme entstehen.

Erst nach diesen Berechnungen kann die Musterzeile folgen, in die die Werte dann eingesetzt werden. Um eine Ausgabe mit zwei Nachkommastellen zu erzeugen, wird die Funktion 'text' benutzt. Deren Ergebnis wird dann rechtsbündig eingesetzt.

Im Nachspann wird dann der summierte Wert mit aufgeschlagener Mehrwertsteuer gedruckt. Die Mehrwertsteuer muß ebenfalls auf zwei Nachkommastellen gerundet werden.

Zur Erweiterung könnten Sie die Bestelldatei noch mit einer Artikeldatei koppeln, die etwa folgende Struktur haben würde:

```
"Artikelnummer"
"Bezeichnung"
"Einzelpreis"
```

In diesem Fall könnten Sie noch jeweils die Artikelbezeichnung in eine Rechnungszeile drucken. Außerdem würde der Preis zentral gespeichert. Eine entsprechende Änderung des Druckmusters sollte Ihnen keine Schwierigkeiten bereiten.

15.2 SPALTENDRUCK

Für manche Anwendungen reicht es nicht aus, wenn die bearbeiteten Sätze jeweils untereinander in der Ausgabe erscheinen. Häufigstes Beispiel dafür ist der Etikettendruck. Hierfür werden vielfach mehrbahnige Formulare eingesetzt.

In diesem Fall müssen die Sätze bis zur gewünschten Anzahl von Spalten nebeneinander gesetzt werden - erst danach wird die nächste Reihe angefangen.

EUDAS unterstützt diese Anwendung. Dazu wird hinter der "%WIEDERHOLUNG"-Anweisung die Anzahl der Spalten als Parameter geschrieben (durch Leerzeichen getrennt). Der Wiederholungsteil wird dann mit der angegebenen Anzahl von Spalten gedruckt. Zu beachten ist, daß Vorspann und Nachspann diese Spaltenanordnung durchbrechen, also immer hinter dem bisher Gedruckten beginnen.

Die Spaltenbreite wird vom Dateilimit des Druckmusters bestimmt. Die Zeilen eines Wiederholungsteils werden bis zum Limit mit Leerzeichen aufgefüllt, wenn der nächste Wiederholungsteil danebengesetzt wird.

Alternativ kann die Spaltenbreite in Zeichen auch als zweiter Parameter angegeben werden. Der Wert gilt jedoch nur für den Wiederholungsteil - Vor- und Nachspann richten sich immer nur nach dem Dateilimit.

Es spielt keine Rolle, ob die nebeneinandergesetzten Wiederholungsteile unterschiedliche Längen haben. Die kürzeren Teile werden einfach bei Bedarf durch Leerzeilen ergänzt. Es ist jedoch zu beachten, daß sich auf diese Weise unterschiedliche Längen für die einzelnen Reihen ergeben können.

Zum Abschluß noch als Beispiel ein Druckmuster für ein dreibahniges Etikettenformular. Die Spaltenbreite und die Länge des Wiederholungsteils richten sich natürlich nach dem verwendeten Formular und müssen im Einzelfall ausprobiert werden.

```
% VORSPANN
#start (1.0, 0.8)#
% WIEDERHOLUNG 3 40

&Vorname %Name
&Strasse

&PLZ %Ort
#free (1.693)#
```

15.3 ANSCHLUSS AN DIE TEXTKOSMETIK

Die Ausgabedatei, die vom Druckgenerator erzeugt wird, kann anschließend noch mit den Programmen der Textkosmetik bearbeitet werden. Der Einsatz von Druckersteuerungskommandos wurde ja schon in Abschnitt 8.4 erwähnt.

Die sinnvollste Anwendung ist die Seitenformatierung mit 'pageform'. Die Kopf-und Fußzeilen können im Vorspann definiert werden; bei Verwendung von Gruppen können diese Zeilen bei jedem Gruppenwechsel neu definiert werden. Auch die Verwendung von Feldmustern in den Kopfzeilen und sogar innerhalb der Textkosmetik-Anweisungen ist kein Problem, da alle diese Anweisungen vom Druckgenerator als ganz normaler Text angesehen werden.

Falls Zeilenwiederholung als Modus eingestellt ist (s. 15.3), kann auch 'lineform' eingesetzt werden - insbesondere bei Proportionalschriften (hier kann der Druckgenerator keinen richtigen Umbruch liefern). Zu diesem Zweck werden die Absatzmarkierungen im Druckmuster direkt in die Ausgabe übernommen. Wird eine Zeile wiederholt, so erhalten jedoch nur die letzte gefüllte Ausgabezeile und eventuell nachfolgende Leerzeilen eine Absatzmarke.

Soll die Druckausgabe in eine bestimmte Datei geschrieben werden, so kann vor Beginn des Druckvorgangs durch das Kommando

```
druckdatei ("Druckdateiname")
```

die Ausgabe in die angegebene Datei geleitet werden. Die Stellung von 'direkt drucken' wird ignoriert. Diese Angabe gilt jedoch nur für einen Druckvorgang; anschließend wird wieder das normale Verfahren angewendet.

Existiert die angegebene Datei schon, wird der Ausdruck am Ende angefügt. Anderenfalls wird die Datei neu eingerichtet.

Für kleinere Systeme ist es vorteilhaft, wenn die Druckausgabe nicht in einer großen Datei gesammelt wird, sondern sich auf mehrere kleine Dateien verteilt. Da die Ausgabedatei nach dem Drucken gelöscht wird, kann man auf diese Weise einen Hintergrundengpaß vermeiden. Es besteht die Möglich-

keit, die maximale Größe der Ausgabedatei mit dem ELAN-Kommando

```
PROC maxdruckzeilen (INT CONST zeilen)
```

auf eine bestimmte Zeilenzahl zu beschränken. Wenn der Druckgenerator nach der Bearbeitung eines Satzes feststellt, daß diese Maximalzahl überschritten wurde, wird die Ausgabedatei direkt gedruckt (falls durch 'direkt drucken' eingestellt) und eine neue Ausgabedatei eröffnet.

Dabei ist jedoch zu beachten, daß Druckersteuerungsanweisungen, die im Vorspann eingestellt wurden, jetzt in der neuen Datei nicht mehr vorhanden sind. In einem solchen Fall würden die folgenden Teile der Ausgabe mit einer anderen Einstellung gedruckt.

Um dies zu vermeiden, können Sie solche Anweisungen in den Initialisierungsteil schreiben. Zeilen im Initialisierungsteil, die keine ELAN-Kommandos oder Anweisungen an den Druckgenerator sind (die also nicht mit "%" oder "%%" beginnen), werden beim Eröffnen einer neuen Ausgabedatei an den Anfang dieser Datei geschrieben.

Druckersteuerungsanweisungen, die ein bestimmtes Schriftbild der Ausgabe erzeugen ('type', 'limit', 'linefeed', 'start' usw.), sollten also in den Initialisierungsteil vor Beginn aller Abschnitte geschrieben werden. Damit werden Sie auch unabhängig von Vorspannwiederholungen bei Gruppenverarbeitung.

15.4 MODI

Gesetzt der Fall, Sie wollen eine Tabelle drucken, deren Einträge auf jeden Fall in voller Länge erscheinen sollen, auch wenn sie die Spaltenbreite überschreiten. Dies würde bedeuten, daß Tabelleneinträge nach rechts geschoben werden, wenn vorhergehende Einträge länger sind. Für diesen Fall können also nur Feldmuster variabler Position (mit "%") eingesetzt werden. Diese werden jedoch auch nach links geschoben, wenn vorher kürzere Inhalte auftreten.

Um dieses Linksschieben zu unterdrücken, können Sie mit folgender Anweisung im Musterteil in den **Tabellenmodus** umschalten:

```
% MODUS 2
```

Der so eingestellte Modus gilt bis zum Ende des jeweiligen Abschnitts. Zu Beginn eines Abschnitts ist der Modus 1 (Normalmodus) eingestellt.

Um diese Anweisung auszuprobieren, sollten Sie folgendes Druckmuster auf unsere Beispieldatei anwenden:

```
% WIEDERHOLUNG
% MODUS 2
&Vorname %Name
```

In der Ausgabe können Sie sehen, daß der Nachname nicht nach links geschoben wird, so daß eine Tabelle entsteht. Ist der Vorname jedoch zu lang, wird die Tabelleneinteilung durchbrochen und der Nachname nach rechts geschoben, um den Vornamen nicht abschneiden zu müssen:

```
Herbert  Wegner
Helga    Sandmann
Albert   Katani
Peter    Ulmen
Karin    Regmann
Hubert   Arken
Anna-Maria Simmern
Angelika Kaufmann-Drescher
Harald   Fuhrmann
Friedrich Seefeld
```

Eine weitere Möglichkeit, überlange Feldinhalte einzusetzen, besteht darin, daß der Rest des Inhaltes, der nicht mehr in den reservierten Raum paßt, in der nächsten Zeile fortgesetzt wird. Dies wird im Modus 3 erreicht. Falls ein Feldinhalt abgeschnitten werden müßte, wird in diesem Modus die gleiche Musterzeile nochmal mit den restlichen Inhalten gedruckt. Dies wird fortgesetzt, bis alle Inhalte abgearbeitet sind.

Damit die Fortsetzung sinnvoll ist, wird das letzte Wort ganz in die nächste Zeile übernommen, falls es zerschnitten würde (ähnlich wie im Editor). Der dadurch freiwerdende Raum in der vorigen Zeile wird mit Leerzeichen gefüllt. Ist rechtsbündiges Einsetzen verlangt, werden die einzelnen Teile jeweils rechtsbündig in ihrem reservierten Platz justiert.

In den Fortsetzungszeilen, werden die Feldmuster, deren Inhalte bereits abgearbeitet sind, leer eingesetzt. Die Mustertexte zwischen den Feldmustern werden in den Fortsetzungszeilen durch Leerzeichen ersetzt.

Die Anzahl der Fortsetzungszeilen kann durch die Anweisung

```
% MEHR n
```

auf eine bestimmte Zahl 'n' festgelegt werden. Wenn alle Inhalte abgearbeitet wurden, aber die Anzahl der Zeilen noch nicht erreicht ist, werden entsprechend viele Zeilen mit leeren Inhalten erzeugt.

Die Zeilenwiederholung kann auch mit dem Tabellenmodus kombiniert werden. Dies wird im Modus 4 erreicht. Felder variabler Position werden auch in diesem Modus nicht nach links geschoben. Außerdem werden aber in Fortsetzungszeilen die Mustertexte zwischen den Feldmustern wiederholt, um z.B. Tabellenbegrenzungen zu erzeugen.

Zum Abschluß dieses Abschnitts eine Zusammenfassung aller möglichen Modi:

Modus	Effekt
1	Normalmodus. "%"-Feldmuster werden auch nach links geschoben. Keine Zeilenwiederholung.
2	Tabellenmodus. "%"-Feldmuster werden nicht nach links geschoben. Keine Zeilenwiederholung.
3	Normalmodus mit Zeilenwiederholung. "%"-Feldmuster werden auch nach links geschoben. Zeilenwiederholung ohne Zwischentexte.
4	Tabellenmodus mit Zeilenwiederholung. "%"-Feldmuster werden nicht nach links geschoben. Zeilenwiederholung mit Zwischentexten.

III.

DETAIL-
BESCHREIBUNG

16
Menüs und Bedienung

EUDAS wird über Menüs bedient. Dabei ist der Bildschirm jeweils in drei Teile aufgeteilt. Der obere Teil enthält ein Menü, in dem mögliche Funktionen aufgeführt sind. In der Mitte erscheint eine Statuszeile, die eine Auswahl der aktuell möglichen Tastendrücke und deren Bedeutung angibt. Der untere Teil ist für den Dialog und die Ausgaben der einzelnen Funktionen vorgesehen.

Mit ESC '?' kann jederzeit eine spezifische Hilfestellung abgerufen werden. Je nachdem, ob gerade eine Funktion ausgeführt wird oder nicht, erscheint diese Hilfestellung im unteren oder im oberen Bildschirmteil.

In EUDAS kann über alle Zustände hinweg gelernt werden. Auch der Editor und alle Dialogeingaben sind über den Lernmodus verbunden. Durch ESC HOP kann man den Lernmodus an beliebiger Stelle einschalten. Alle nachfolgenden Tastendrücke werden gelernt. Mit ESC HOP 'Taste' kann man diese Sequenz dann auf eine Taste legen. Die Sequenz kann dann durch ESC 'Taste' ausgeführt werden. Ob der Lernmodus eingeschaltet ist, kann jedoch nur im Editor angezeigt werden.

16.1 MENÜS

Menüs dienen zur Auswahl von Funktionen am Bildschirm. Sie werden im oberen Teil des Bildschirms angezeigt. Mögliche Funktionen sind durch ein vorangehendes Minuszeichen gekennzeichnet. Dieses Minuszeichen kann mit Hilfe der Cursortasten angefahren werden. Die Leertaste führt die Funktion aus, auf der der Cursor steht.

Folgende Tasten(kombinationen) dienen der Cursorbewegung:

RECHTS, LINKS, OBEN, UNTEN:
Bewegung des Cursors in die entsprechende Richtung, falls in dieser Richtung ein Minuszeichen steht.

HOP RECHTS, HOP LINKS, HOP OBEN, HOP UNTEN:
Bewegung des Cursors auf das letzte Minuszeichen in der entsprechenden Richtung.

HOP HOP:
Bewegung des Cursors auf das erste Minuszeichen in der ersten Zeile, die ein Minuszeichen enthält.

TAB:
Zum nächsten Minuszeichen in der Zeile, sonst auf das erste in der nächsten Zeile.

RETURN:
Zum ersten Minuszeichen in der nächsten Zeile.

Durch ESC '?' wird eine Hilfestellung gezeigt, die sich auf die Funktion bezieht, auf der sich der Cursor gerade befindet und die das Menü zeitweise überschreibt. Mit ESC ESC kann ein Kommandodialog geführt werden. Mit ESC 'k' wird der Cursor auf die zuletzt aufgerufene Funktion positioniert. ESC 'q' beendet das Menü.

Die weiteren Tastendrücke in der Zusammenfassung:

LEERTASTE:
Führt die Funktion aus, vor der der Cursor steht. Das Minuszeichen wird in ein Pluszeichen verwandelt, um die Funktion zu kennzeichnen, die gerade ausgeführt wird.

ESC HOP:
Schaltet den Lernmodus ein.

ESC HOP 'Taste':
Legt bei eingeschaltetem Lernmodus die gelernte Sequenz auf die entsprechende Taste.

ESC ESC:
Führt den Kommandodialog in der Zeile, in der sich der Cursor befindet. Bevor das ELAN-Kommando ausgeführt wird, wird der Cursor in den unteren Bildteil bewegt. Vorsicht bei Funktionen, die den Bildschirm löschen und damit das Menü überschreiben. Mit ESC 'k' kann im Kommandodialog das letzte Kommando wiedergeholt werden.

ESC '?':
Ruft eine Hilfestellung auf, die sich auf die Funktion bezieht, auf der der Cursor gerade steht.

ESC 'k':
Positioniert den Cursor auf die zuletzt ausgeführte Funktion.

Tritt bei der Ausführung einer Funktion ein Fehler auf, so wird dieser im unteren Bildschirmteil ausgegeben. Anschließend steht der Cursor wieder auf der letzten Funktion, die den Fehler verursacht hat, damit diese eventuell gleich wieder ausgeführt werden kann.

Auch wenn das Menü noch nicht komplett ausgegeben wurde, werden bereits Tastendrücke angenommen und dekodiert. Es ist also möglich, eine Funktion anzuwählen und auszuführen, ohne daß das Menü überhaupt ausgegeben wird. Dies ist insbesondere bei der Ausführung von Lernsequenzen zu beachten.

Die einzelnen Funktionen können auch durch Drücken des Buchstabens ausgeführt werden, der vor dem Minuszeichen steht. Der entsprechende Buchstabe kann groß oder klein angegeben werden.

16.2 AUSWAHL

Die Auswahlfunktion dient dazu, aus vorhandenen Datei- oder Feldnamen in bestimmter Reihenfolge auszuwählen. Die einzelnen Namen werden untereinander aufgelistet. Vor jedem Namen ist ein 'o' zum Ankreuzen angegeben. Mit den Cursortasten kann der Cursor vor einen bestimmten Namen positioniert werden. Mit 'x' kann dieser Name dann angekreuzt werden. Das Ankreuzen kann durch 'o' wieder rückgängig gemacht werden.

Die Reihenfolge des Ankreuzens wird durch vorangestellte Nummern gekennzeichnet. Die Namen werden von der entsprechenden Funktion später in genau dieser Reihenfolge verwendet.

Wenn nicht alle Namen auf den Bildschirm passen, kann die Darstellung gerollt werden. Ein Teil der Überschrift bleibt dabei stehen; am Anfang und am Ende wird jeweils eine Abschlußzeile zur Kennzeichnung mitgerollt.

Mit ESC '?' kann eine Hilfestellung abgerufen werden. Mit ESC 'q' wird die Auswahl beendet. Mit ESC 'a' können die Auswahl und die in Ausführung befindliche Operation abgebrochen werden.

Die möglichen Tasten(kombinationen) im einzelnen:

RECHTS, LINKS, OBEN, UNTEN:
Cursorbewegung in die entsprechende Richtung, falls möglich. RECHTS und LINKS funktionieren natürlich nur bei mehrspaltiger Auswahl (z.Zt. nicht verwendet).

HOP OBEN, HOP UNTEN:
Bewegt den Cursor in die erste bzw. letzte Zeile, wenn er sich nicht dort befindet. Anderenfalls wird um eine Seite gerollt, falls möglich. Anfang und Ende der Auswahl sind durch eine Abschlußzeile markiert.

HOP RETURN:
Macht die aktuelle Zeile zur ersten Zeile.

HOP HOP:
Zum ersten Namen auf der Seite.

'x', '+':
Ankreuzen eines Namens.

'o', '-':
Ankreuzung löschen. Die Nummern werden entsprechend korrigiert.

HOP 'x', HOP '+':
Alle noch nicht angekreuzten Namen ankreuzen. Dabei wird von oben nach unten durchnumeriert.

HOP 'o', HOP '-':
Alle Ankreuzungen löschen.

LEERTASTE:
Aktuellen Namen ankreuzen und Auswahl verlassen (falls nur ein Name gewünscht wird).

ESC '1':
Zum ersten Namen. Gegebenenfalls wird gerollt.

ESC '9':
Zum letzten Namen. Gegebenenfalls wird gerollt.

ESC HOP:
Lernmodus einschalten.

ESC HOP 'Taste':
Lernsequenz auf Taste legen.

ESC '?':
Hilfestellung abrufen.

ESC 'a':
Auswahl und Funktion abbrechen.

ESC 'q':
Auswahl beenden.

16.3 HILFE UND DIALOG

In den meisten Situationen kann durch ESC '?' eine spezifische Hilfestellung abgerufen werden. Die Anzeige der Hilfsinformation geschieht je nach Stelle des Aufrufs im oberen oder unteren Bildschirmteil.

Die Texte sind seitenweise aufgebaut. Es wird immer eine Seite angezeigt. Mit ESC 'w' bzw. ESC 'z' kann auf die nächste bzw, vorige Seite umgeblättert werden. Mit ESC 'q' wird die Hilfestellung wieder verlassen und die Situation wiederhergestellt, in der die Hilfe aufgerufen wurde.

Die meisten Funktionen wickeln zur Eingabe von zusätzlichen Parametern oder zum Stellen von Fragen einen Dialog in der unteren Schirmhälfte ab. Es gibt zwei Möglichkeiten des Dialogs: eine Frage oder die Eingabe eines Textes.

Bei einer Frage kann man mit 'j' oder 'n' antworten. Sowohl große als auch kleine Buchstaben werden akzeptiert. Mit ESC '?' kann eine Hilfsinformation zu der Frage abgerufen werden. ESC 'a' bricht die fragende Funktion ab. Zu beachten ist, daß vor Fragen der Tastenpuffer gelöscht wird und damit ein Lernen über eine Frage hinweg unmöglich ist.

Bei der Eingabe eines Textes können die üblichen Operationen zum Editieren in einer Zeile verwendet werden. Die Eingabe wird durch RETURN beendet. Auch hier kann durch ESC '?' eine Hilfsinformation abgerufen werden. ESC 'a' bricht ebenfalls die fragende Funktion ab. In einigen Fällen (ersichtlich aus der Statuszeile) kann durch ESC 'z' eine Auswahl der verfügbaren Namen abgerufen werden.

16.4 MENÜSTRUKTUR

EUDAS verfügt über fünf Menüs. Das Hauptmenü erscheint nach dem Aufruf der Prozedur 'eudas'.

```
:--------------------- ( EUDAS - Hauptmenue ) ----------------------------:
: a - aktuelle Datei ansehen        b - aktuelle Datei bearbeiten         :
: o - neue Datei oeffnen            f - Feldstruktur aendern              :
: k - Dateien koppeln               r - EUDAS-Datei sortieren             :
: s - aktuelle Dateien sichern      v - Dateiverwaltung                   :
:-----------------------------------------------------------------------:
```

Die Auswahlen "aktuelle Datei ansehen", "aktuelle Datei bearbeiten" und "Dateiverwaltung" führen in ein Untermenü. Die übrigen Funktionen auf der linken Reihe sind in Kapitel 17 beschrieben, die Funktionen auf der rechten Reihe in Kapitel 20.

```
:------------------------- << Ansehen >> ---------------------------------:
: w - Satz weiter                   z - Satz zurueck                      :
: s - Selektion einstellen          l - Selektion loeschen                :
: d - direkt auf Satz               v - Bild vertikal verschieben         :
: e - Einzelsatz bearbeiten         b - aktuelle Datei bearbeiten         :
:-----------------------------------------------------------------------:
```

In diesem Menü erscheint im unteren Teil des Bildschirms ein Teil des Dateiinhalts der aktuellen Datei. Wenn keine Datei geöffnet ist, läßt sich dieses Menü nicht anwählen. Die Funktionen "Einzelsatz bearbeiten" und "aktuelle Datei bearbeiten" führen in weitere Menüs, die übrigen Funktionen sind in Kapitel 18 beschrieben.

```
:--------------------- <<< Einzelsatz bearbeiten >>> ----------------------:
: e - neuen Satz einfuegen          a - Satz aendern                      :
: m - Markierung aendern            l - alle Markierungen loeschen        :
: t - Satz tragen                   h - Satz holen                        :
: d - Darstellung umschalten        f - Feldauswahl                       :
:-----------------------------------------------------------------------:
```

Auch in diesem Menü wird ein Dateiausschnitt angezeigt. Die Funktionen "Satz tragen" und "Satz holen" sind in Kapitel 20 beschrieben, der Rest in Kapitel 18.

```
:-------------------- << aktuelle Datei bearbeiten >> ---------------------:
: d - nach Muster drucken          t - Saetze tragen                      :
: e - Textdatei erstellen/aendern  k - Saetze kopieren                    :
: u - Textdatei drucken            n - nach Vorschrift aendern            :
: a - aktuelle Datei ansehen       v - Dateiverwaltung                    :
:-------------------------------------------------------------------------:
```

Dieses Menü bearbeitet die aktuelle Datei als Ganzes. Es kann nicht aufgerufen werden, wenn keine Datei geöffnet ist. Die Funktionen "aktuelle Datei ansehen" und "Dateiverwaltung" führen in andere Menüs. Die übrigen Funktionen auf der linken Seite sind in Kapitel 19 beschrieben, die Funktionen auf der rechten Seite in Kapitel 20.

```
:---------------------- << Dateiverwaltung >> ------------------------------:
: s - Datei auf Archiv sichern     r - Datei reorganisieren                :
: h - Datei von Archiv holen       m - Datei umbenennen                    :
: o - auf Archiv loeschen          l - Datei loeschen                      :
: a - Archivuebersicht             u - Dateiuebersicht                     :
:-------------------------------------------------------------------------:
```

Die Funktionen dieses Menüs sind im folgenden Abschnitt beschrieben, da Sie keine EUDAS-spezifischen Funktionen darstellen.

16.5 DATEIVERWALTUNG

Im Menü "Dateiverwaltung" sind allgemeine Funktionen aufgeführt, die Dateien betreffen und für die Benutzung von EUDAS gebraucht werden. Diese Funktionen sind ein Ersatz für sonst im Monitor gebräuchliche Prozeduren.

Vier Funktionen befassen sich mit Dateien auf dem Archiv. Die Funktion

- Archivuebersicht

liefert eine Übersicht aller auf dem eingelegten Archiv vorhandenen Dateien. Als Besonderheit ist zu vermerken, daß der Archivname nicht eingegeben werden muß, sondern vom Programm selbst bestimmt wird. Ansonsten ist diese Funktion identisch mit dem Kommando 'list (archive)'. Die Übersicht muß mit ESC 'q' verlassen werden.

Bei der Funktion

- Datei von Archiv holen

muß ebenfalls kein Archivname angegeben werden. Nur der gewünschte Dateiname wird erfragt. Ist der Name nicht genau bekannt oder sollen mehrere Dateien geholt werden, kann mit ESC 'z' eine Auswahl der vorhandenen Dateien angefordert werden. Die dort angekreuzten Dateien werden dann nacheinander vom Archiv geholt.

Bei den beiden anderen Funktionen muß jeweils der Archivname eingegeben werden, damit die Dateien nicht auf ein falsches Archiv geschrieben werden. Zum Ändern angeboten wird dort jedoch der zuletzt verwendete Archivname (und nicht etwa der Name des eingelegten Archivs).

Die Funktion

- Datei auf Archiv sichern

erwartet anschließend einen Dateinamen. Wird hier ESC 'z' eingegeben, können von den auf dem System befindlichen Dateien einige ausgewählt werden, die dann in der spezifizierten Reihenfolge auf die Archivdiskette geschrieben werden. Existiert eine zu sichernde Datei bereits, wird vor dem Überschreiben gefragt. Wird diese Frage dann verneint, wird die Operation mit der nächsten Datei fortgesetzt.

Die Funktion

- Datei auf Archiv loeschen

löscht eine Datei auf dem Archiv. Auch hier kann bei der Eingabe des Dateinamens mit ESC 'z' eine Auswahl der auf dem Archiv vorhandenen Dateien angefordert werden. Vor dem Löschen jeder einzelnen Datei wird nochmal zur Sicherheit gefragt.

Die vier Funktionen auf der rechten Seite befassen sich mit Dateien im System. Die Funktion

- Dateiübersicht

liefert eine Übersicht der Dateien wie bei 'list'. Da die Übersicht den Editor aufruft, muß die Übersicht mit ESC 'q' verlassen werden. Die Funktion

- Datei loeschen

erwartet einen Dateinamen. Durch ESC 'z' kann eine Auswahl der vorhandenen Dateien angefordert werden, in der die zu

löschenden Dateien angekreuzt werden können. Bevor eine Datei gelöscht wird, wird zur Sicherheit noch gefragt.

Die Funktion

- Datei umbenennen

erfragt auch einen Dateinamen (auch hier ESC 'z' möglich). Der Name der ausgewählten Datei wird dann zum Ändern angeboten.

Mit der Funktion

- Datei reorganisieren

können eine einzelne Datei bzw. eine Auswahl von Dateien reorganisiert werden. Dabei wird bei EUDAS-Dateien die entsprechende Reorganisationsprozedur aufgerufen.

17
Das virtuelle Dateimodell

17.1 ÖFFNEN

Die meisten EUDAS-Funktionen arbeiten nicht direkt auf einer EUDAS-Datei, sondern auf der sogenannten **virtuellen Datei**, die aus mehreren realen Dateien bestehen kann. Die virtuelle Datei erscheint nach außen hin wie eine einzelne EUDAS-Datei. Die Operationen auf der virtuellen Datei werden jeweils auf die einzelnen Bestandteile abgebildet.

Damit eine EUDAS-Datei Bestandteil der virtuellen Datei wird, muß sie geöffnet werden. Dieses Öffnen kann auf dreierlei Art und Weise geschehen.

Das Öffnen der ersten Datei stellt eine neue virtuelle Datei her. Die Feldnamen und Feldeigenschaften der ersten Datei werden in der virtuellen Datei übernommen. Dies ist der Normalfall, in dem sich die virtuelle Datei noch nicht von der zugrundeliegenden Datei unterscheidet.

```
.......................................................
. +---------+  + - - - - - +  + - - - - - +
. !  EUDAS- !  ! gekettete !  ! gekettete !
. ! Datei 1 !  !  Datei A  !  !  Datei B  !
. +---------+  + - - - - - +  + - - - - - +
.......................................................
        .              .                  Hauptdatei
        .              .
  + - - - - - +        .
  ! gekoppelte!        .
  !  Datei C  !        .
  + - - - - - +        .
                + - - - - - +
                ! gekoppelte!
                !  Datei D  !
                + - - - - - +
```

Schematischer Aufbau der virtuellen Datei

Weitere Dateien können gekettet oder gekoppelt werden. Gekettete Dateien werden logisch an die zuerst geöffnete Datei angehängt. Ihre Dateistruktur wird ignoriert, sollte aber mit der ersten Datei übereinstimmen. Die Folge aneinander geketteter EUDAS-Dateien wird als **Hauptdatei** bezeichnet.

Das Öffnen und Ketten geschieht mit der Menüfunktion

- neue Datei oeffnen

Zuerst wird gegebenenfalls das vorherige Dateimodell gesichert (s. 17.4). Anschließend wird gefragt, ob die Datei verändert werden soll oder nicht. Falls auch geändert werden soll, wird von den einzelnen Dateien eine Sicherheitskopie gemacht.

Danach kann ein Dateiname eingegeben werden. Sollen Dateien gekettet werden, müssen Sie an dieser Stelle die Dateiauswahl abrufen. Die zuerst angekreuzte Datei wird dann geöffnet, die weiteren dazugekettet.

Falls der eingegebene Dateiname noch nicht existiert, wird gefragt, ob eine neue Datei eingerichtet werden soll. Beim Einrichten werden dann die gleichen Eingaben erwartet wie beim Ändern der Feldstruktur (s. 20.1).

In der Hauptdatei werden die Sätze von 1 an durchnumeriert; die Aufeinanderfolge der Sätze wird durch die Anordnung der Sätze in den einzelnen Dateien und die Reihenfolge bestimmt, in der die Dateien gekettet wurden.

Die gekoppelten Dateien werden der Hauptdatei untergeordnet. Die in ihnen enthaltenen Informationen werden nur angezeigt, wenn sie mit einem Satzinhalt der Hauptdatei korrespondieren. Der Mechanismus dieser Satzkopplung wird im nächsten Abschnitt beschrieben.

17.2 KOPPELN

Dateien werden gekoppelt mit der Funktion

- Dateien koppeln

Dabei kann entweder ein Dateiname angegeben oder mehrere Dateien ausgewählt werden. Die zu koppelnden Dateien müssen bereits existieren. Ob Ändern erlaubt ist, richtet sich nach der Einstellung beim Öffnen.

Die Sätze der gekoppelten Dateien werden in Relation zu den Sätzen in der Hauptdatei gesetzt. Zu jedem Satz in der Hauptdatei kann eine Anzahl von Sätzen aus jeder Koppeldatei gehören. Diese Sätze müssen in den Inhalten der sogenannten Koppelfelder übereinstimmen.

Welche Felder Koppelfelder sind, richtet sich nach den Feldnamen. Die ersten Felder der Koppeldatei, die auch in der

Hauptdatei vorhanden sind, werden als Koppelfelder betrachtet. Die Koppelfelder müssen also bei der Koppeldatei am Anfang stehen - in der Hauptdatei kann jedes beliebige Feld ein Koppelfeld sein.

Wenn eine Datei zur virtuellen Datei gekoppelt wird, werden alle Felder, die nicht Koppelfelder sind, in die virtuelle Datei aufgenommen. Die Koppelfelder brauchen nicht noch mal wiederholt zu werden, da ihr Inhalt ja immer identisch ist. Zu beachten ist, daß bei diesem Verfahren auch Namenskonflikte entstehen können, wenn nach den Koppelfeldern später wieder ein Feldname vorkommt, der auch in der Hauptdatei vorhanden ist.

```
                x1
                x2
Hauptdatei      A -`            x1
                x3 !            x2
                B -+-`          A
                x4 ! !    ==>   x3    virtuelle Datei
                   ! !          B
                A -' !          x4
Koppeldatei     B ---'          y1
                y1              y2
                y2
```

Beim Vorwärtsgehen in der virtuellen Datei werden zu jedem Satz der Hauptdatei nacheinander alle möglichen Kombinationen der zugehörigen Koppelsätze angezeigt. Die Satznummer bleibt dabei gleich; die einzelnen Kombinationen werden von 1 an durchgezählt. Beim Rückwärtsgehen wird aus technischen Gründen immer nur die erste Kombination angezeigt.

Existiert zu einem Satz kein passender Koppelsatz, so bleiben die entsprechenden Felder leer. Die Koppelsätze müssen in der ganzen Koppeldatei gesucht werden, daher ist bei großen Koppeldateien die Suchzeit zu berücksichtigen.

17.3 ÄNDERUNGEN

In der virtuellen Datei kann auch geändert werden. Dabei ist jedoch Vorsicht angebracht. Es muß festgelegt sein, wie Änderungen der einzelnen Felder auf die beteiligten Dateien abgebildet werden.

Falls die virtuelle Datei keine Koppeldateien enthält, werden Änderungen am aktuellen Satz an der zugehörigen Datei durchgeführt. Das Löschen eines Satzes wird auch direkt

in der Datei durchgeführt. Ein neuer Satz wird immer in der Datei eingefügt, zu der der aktuelle Satz gehört - am Ende der ersten Datei kann also kein Satz eingefügt werden, wenn noch weitere Dateien folgen.

Enthält die virtuelle Datei Koppeldateien, werden die Änderungen in der Hauptdatei wie oben beschrieben durchgeführt. Änderungen in den Koppeldateien werden nach folgender Entscheidungstabelle behandelt:

	1	2	3	4	5
Koppelfelder verändert	N	J	J	N	N
Übrige Felder verändert	N	-	-	J	J
Übrige Felder leer	-	J	N	-	N
Vorher Koppelsatz vorhanden	-	-	-	J	N
Neuen Satz einfügen			x		x
Koppelsatz ändern				x	
Kopplung aktualisieren		x			

Fall 1: Es wurden keine Veränderungen an den Feldern des Koppelsatzes vorgenommen, daher ist auch keine Aktion notwendig.

Fall 2: Eines der Koppelfelder wurde verändert. Die Änderung wird in der Hauptdatei durchgeführt. Die übrigen Felder des Koppelsatzes sind jedoch als leer angegeben. In diesem Fall wird der Koppelsatz nicht verändert, sondern nur eine neue Korrespondenz gesucht.

Fall 3: Eines der Koppelfelder wurde verändert, gleichzeitig enthalten aber auch die anderen Felder Informationen. In diesem Fall wird ein neuer Satz in der Koppeldatei angefügt, der die neuen Inhalte enthält. So wird vermieden, daß an anderer Stelle plötzlich kein passender Koppelsatz mehr vorhanden ist.

Fall 4: Nur Felder der Koppeldatei, die nicht Koppelfelder sind, wurden verändert, außerdem existierte ein korrespondierender Satz in der Koppeldatei. In diesem Fall werden die Informationen im Koppelsatz abgeändert.

Fall 5: Wie 4, nur war vorher noch kein Koppelsatz vorhanden (Felder waren leer). In diesem Fall muß ein neuer Satz in die Koppeldatei eingefügt werden.

Einfügungen in die Koppeldatei geschehen immer am Dateiende.

Beim Löschen eines Satzes der virtuellen Datei bleiben die Koppeldateien unverändert. Nach dem Einfügen eines neuen Satzes wird nur dann ein Satz in einer Koppeldatei eingefügt, wenn dieser Satz nicht nur Koppelfelder enthalten würde. Falls beim Einfügen nur die Koppelfelder angegeben werden, wird ein korrespondierender Satz in der Koppeldatei gesucht. Vergleichen Sie hierzu die Regeln beim Ändern.

17.4 SICHERN

Falls Änderungen der virtuellen Datei erlaubt sind, arbeitet EUDAS immer auf Sicherheitskopien der beteiligten Dateien. Eine Datei, die wirklich verändert wurde, muß vor dem Aufbau einer neuen virtuellen Datei gesichert oder explizit gelöscht werden.

Die Sicherung erfolgt durch die Funktion

- aktuelle Dateien sichern

Falls keine Änderungen erlaubt waren, ist eine Sicherung nicht notwendig, da auch keine Kopien angelegt wurden. Anderenfalls wird für jede offene Datei angegeben, ob sie verändert wurde oder nicht. Außerdem wird gefragt, ob die Datei gesichert werden soll.

Wird die Frage bejaht, wird der alte Name der Datei zum Ändern angeboten. An dieser Stelle kann ein neuer Name für die Datei angegeben werden, so daß die alte Version nicht überschrieben wird. Wenn der angegebene Dateiname bereits existiert, wird vor dem Überschreiben noch nachgefragt.

Nachdem alle Dateien behandelt wurden, wird gefragt, ob die Kopien gelöscht werden sollen. Wenn die Kopien nicht gelöscht werden, bleiben die Dateien weiterhin offen. Anderenfalls gehen Kopien verloren, die geändert waren, aber nicht gesichert wurden.

Falls alle Dateien entweder gesichert oder nicht verändert sind, werden beim nächsten Öffnen einer neuen virtuellen Datei die vorherigen Dateikopien gelöscht.

18
Anzeigen und Ändern

Die Funktion

- aktuelle Datei anzeigen

im Hauptmenü führt in ein weiteres Menü, in dem im unteren Teil des Bildschirms immer ein Ausschnitt der virtuellen Datei zu sehen ist. Das Anzeigemenü hat ein weiteres Untermenü, das über

- Einzelsatz bearbeiten

aufgerufen wird. Auch in diesem Menü erfolgt eine Anzeige in der unteren Bildschirmhälfte.

Mit den Funktionen

- Satz weiter

und

- Satz zurueck

kann man sich in der Datei bewegen. Ist eine Selektionsbedingung eingestellt (s. 18.3), wird immer der nächste ausgewählte Satz angezeigt. Ausnahme: der erste Satz bzw. das Dateiende werden angezeigt, wenn kein weiterer ausgewählter Satz mehr vorhanden ist. Durch die Funktion

- direkt auf Satz

kann man auf einen bestimmten Satz positionieren. Die Nummer des Satzes wird erfragt.

18.1 ANZEIGEFORMAT

Die Anzeige und Eingabe von Einzelsätzen sowie die Eingabe von Suchmustern geschieht in einem Standardformular in einem rechteckigen Fenster. Dieses Fenster befindet sich in der Regel in der unteren Bildschirmhälfte.

Das Formular besteht aus vier Teilen: der Überschrift, den Feldnamen, den Feldinhalten und der Abschlußzeile (s. Abbildung).

```
                              Überschrift
Satz 33      ..SUCH+..MARK-................ Zeile 1
Feld 1
Feld 2
Feld 3              Feldinhalte

Feld 4
.....................................................
Feldnamen                        Abschlußzeile
```

Die Überschrift zeigt folgende Informationen an:

Satz n[-m]
Die Satznummer des aktuellen Satzes, bei gekoppelten Dateien auch die Satzkombination. Die Satznummer wird am linken Rand dargestellt. Am Ende der Datei erscheint in diesem Feld "ENDE".

SUCH+/-
Zeigt an, ob der aktuelle Satz die eingestellte Suchbedingung erfüllt oder nicht (wird während Eintragungen nicht angezeigt).

MARK+/-
Zeigt an, ob der aktuelle Satz markiert ist oder nicht (Wird während Eintragungen nicht angezeigt).

Zeile n
Zeilennummer des obersten angezeigten Feldes (bei Anzeige) bzw. der aktuellen Cursorzeile (während Eintragungen).

Die Feldnamen sind zur Unterscheidung von den Feldinhalten markiert dargestellt. Die Breite der Feldnamen richtet sich nach der Länge des längsten Feldnamens. Ist dieser zu lang, um noch eine ordentliche Anzeige zu ermöglichen, wird bei einer bestimmten Länge der Rest des Namens abgeschnitten.

Zwischen dem Feldnamen an der linken Seite und dem danebenstehenden Feldinhalt besteht immer eine Korrespondenz, d.h. der Inhalt eines Feldes wird direkt neben dem Namen dargestellt. In der Regel wird pro Feld eine Bildschirmzeile reserviert. Kann der Feldinhalt jedoch nicht mehr in einer Zeile untergebracht werden, werden weitere Zeilen zur Darstellung dieses Feldes herangezogen. In diesen Zeilen steht statt des Feldnamens nur ein markierter Leerraum.

Alle folgenden Zeilen ohne Namen gehören zu dem gleichen Feld. Der Inhalt wird auf diese Zeilen umbrochen, d.h. wenn ein Wort nicht mehr auf die Zeile paßt, wird es komplett in die nächste Zeile geschrieben (wie beim Editor). Wörter werden nur dann zerschnitten, wenn sie nicht als Ganzes auf eine Zeile passen. Wörter werden untereinander durch Leerzeichen getrennt.

Aus Effizienzgründen werden in bestimmten Fällen auch mehr Folgezeilen als nötig angezeigt. Hat nämlich ein neuer Satz einen kürzeren Inhalt als der vorige, so werden die Feldnamen nur dann wieder zusammengerückt, wenn das ganze Bild neugeschrieben werden muß. Anderenfalls werden nur die Feldinhalte aktualisiert.

Die Bildausgabe wird unterbrochen, wenn 'w' oder 'z' gedrückt wurde, da dann die Inhalte des aktuellen Satzes nicht mehr interessieren.

Rollen

Da nicht alle Felder auf den Bildschirm passen müssen, kann das Bild gerollt werden. Dies geschieht mit der Auswahl

- Bild vertikal verschieben

Dadurch gelangt man in einen Zustand, in dem Tastenkombinationen zum vertikalen Verschieben der Anzeige eingegeben werden können. Mit ESC 'q' kommt man aus diesem Zustand zurück ins Hauptmenü.

Mit ESC UNTEN wird um eine Seite nach unten geblättert, mit ESC OBEN wieder zurück. Hinter dem letzten Feld erscheint ein markierter Balken als Abschlußzeile. Weiter als bis zum Erscheinen dieses Balken kann nicht gerollt werden. Mit ESC '1' wird ganz an den Anfang gerollt, mit ESC '9' ganz ans Ende.

Bei Feldern, die sich über mehrere Zeilen erstrecken, kann es passieren, daß nach dem Rollen die erste Bildschirmzeile nicht die erste Zeile eines Feldes ist, also der erste Teil eines Feldes nicht dargestellt wird. Trotzdem wird in diesem Fall in der ersten Anzeigezeile der Feldname angezeigt.

Übersichtsanzeige

Im Gegensatz zur normalen Anzeige, bei der ein Satz pro Bildschirm dargestellt wird, können mit der Übersichtsanzeige mehrere Sätze gleichzeitig überschaut werden. Dabei wird jeder Satz in einer Zeile untergebracht. Die Darstellung kann mit der Funktion

- Darstellung umschalten

jeweils zwischen Standardformular und Übersichtsanzeige umgeschaltet werden.

In jeder Zeile steht die Nummer des jeweiligen Satzes, eine Anzeige, ob er markiert ist (+) oder nicht (-) und die Feldinhalte in der gewählten Reihenfolge und Auswahl, jeweils duch Komma und Leerzeichen getrennt. Inhalte, die nicht mehr auf die Zeile passen, werden abgeschnitten.

Es werden nur durch das Suchmuster ausgewählte Sätze angezeigt (s. 18.3). Ist der aktuelle Satz nicht ausgewählt, so erscheint an seiner Stelle "<< >>" als Hinweis. In der Überschrift sind die Feldnamen angegeben - durch Komma getrennt, so viele wie hinpassen.

Die Satznummer des aktuellen Satzes ist jeweils markiert. Blättern ist möglich mit der Funktion

- Bild vertikal verschieben

ESC OBEN und ESC UNTEN verschieben um jeweils eine Seite, OBEN und UNTEN um jeweils einen Satz. Im letzteren Fall wird der Bildschirm erst dann gerollt, wenn der aktuelle Satz nicht mehr im Bildausschnitt liegen würde.

Zum Einfügen, Ändern und Suchen wird automatisch zeitweise in die Normaldarstellung umgeschaltet.

Feldauswahl

Man kann auswählen, welche Felder in welcher Reihenfolge angezeigt werden sollen. Dies dient der besseren Übersicht. Von der Anzeige werden nur die ausgewählten Felder behandelt, die anderen Felder bleiben leer, werden nicht verändert oder berücksichtigt. Die Anzeigeauswahl ändert jedoch nichts an der Dateistruktur.

Die Auswahl wird mit der Funktion

- Feldauswahl

durchgeführt. Es werden alle Felder angeboten. Die anzuzeigenden Felder können in der gewünschten Reihenfolge angekreuzt werden.

Die Auswahl wird separat für beide Darstellungsarten gespeichert. Bei einem neuen Öffnen werden wieder alle Felder angezeigt.

18.2 SATZEDITOR

Durch die Funktion

- Satz aendern

können die Inhalte des aktuellen Satzes verändert werden. Die Funktion

- Satz einfuegen

fügt einen leeren Satz vor dem aktuellen Satz ein und ermöglicht die Eingabe der Inhalte. Zur Auswirkung von Änderungen auf gekoppelte Dateien s. 17.3.

Beim Ändern, Einfügen und Eingeben des Suchmusters wird im EUDAS-Formular der Editor aufgerufen. Das Editorfenster beschränkt sich auf den rechten Teil des Formulars, der einen Teil der Überschrift und die Feldinhalte umfaßt. Im Editor können dann die entsprechenden Inhalte eingegeben bzw. verändert werden.

Vorsicht! Das Editorfenster kann gegenüber den Feldnamen verschoben werden (durch Rollen im Editor). Dabei geht die sichtbare Korrespondenz zwischen Feldnamen und Feldinhalten verloren. Ein solcher Fall wird durch einen markierten Balken mit entsprechender Meldung angezeigt. Durch ESC '1' wird das Fenster aber wieder zurechtgerückt.

Aus diesem Grund sollte im Satzeditor auf HOP OBEN und HOP UNTEN sowie auf RETURN am Ende des Fensters verzichtet werden. Auch HOP RUBIN und HOP RUBOUT sollten nicht verwendet werden, weil auf diese Weise die Anzahl der Zeilen verändert wird. Eine solche Störung kann nicht durch ESC '1' beseitigt werden. Von Hand müssen die entsprechenden Zeilen wieder gelöscht oder eingefügt werden.

Rollen unter Beibehaltung der Korrespondenz ist durch ESC OBEN und ESC UNTEN möglich. Diese Funktionen wirken wie bei der Anzeige - das Editorfenster wird ebenfalls entsprechend mitgerollt. Mit ESC '1' kann wie üblich auf die erste, mit ESC '9' auf die letzte Zeile gesprungen werden. Auch diese Funktionen passen die Feldnamen entsprechend an.

Falls die für ein Feld vorgesehenen Zeilen nicht für den Inhalt ausreichen, kann durch ESC RUBIN eine weitere Zeile für dieses Feld bereitgestellt werden. ESC RUBIN wirkt wie zweimal HOP RUBIN, die Korrespondenz mit den Feldnamen bleibt jedoch gewahrt.

Zum Löschen steht ESC RUBOUT zur Verfügung. Es löscht eine ganze Zeile, aber nur wenn für dieses Feld noch andere Zeilen zur Verfügung stehen, wird die Zeile tatsächlich vom Bildschirm entfernt. Im Normalfall bleibt sonst eine Leerzeile für dieses Feld stehen.

Ist die Information für ein Feld auf mehrere Zeilen verteilt, so werden diese Zeilen zur Verarbeitung aneinandergehängt. Gegebenenfalls wird zwischen zwei Zeilen noch ein Leerzeichen eingefügt.

Der Editor kann wie üblich mit ESC 'q' verlassen werden. ESC 'a' bricht die Funktion ab, ohne die erfolgten Änderungen und Eingaben zu berücksichtigen. Mit ESC 'w' und ESC 'z' kann das Bearbeiten von mehreren Sätzen beschleunigt werden. Durch diese Tastenkombinationen wird der Editor verlassen und die gleiche Operation (Ändern/Einfügen) beim nächsten bzw. vorigen Satz wiederholt.

18.3 SUCHMUSTER

Durch die Funktion

- Selektion einstellen

wird eine Auswahlbedingung in Form eines Suchmusters eingestellt. Diese Bedingung hat Auswirkungen auf weitere Funktionen: die Anzeige und alle Verarbeitungsfunktionen beachten nur Sätze, die die Suchbedingung erfüllen. In der Anzeige erscheint "SUCH", wenn ein Suchmuster eingestellt ist. Ein '+' dahinter zeigt an, daß die Suchbedingung auf den angezeigten Satz zutrifft.

Mit der Funktion

- Selektion loeschen

kann die Auswahlbedingung wieder abgeschaltet werden.

Das Suchmuster gibt für jedes Feld bestimmte Bedingungen an. Es wird im Standardformular mit Hilfe des Satzeditors eingegeben. Dabei stehen neben jedem Feld die Bedingungen für dieses Feld in einer intuitiv verständlichen Form.

Folgende Einzelbedingungen sind möglich:

Formulierung	Bedingung
	keine Bedingung
Muster	Inhalt ist gleich Muster
Muster..	Inhalt ist größergleich Muster
..Muster	Inhalt ist kleiner Muster
Muster1..Muster2	Inhalt liegt dazwischen
*Muster	Inhalt endet mit Muster
Muster*	Inhalt beginnt mit Muster
Muster	Inhalt enthält Muster
*	Inhalt ist nicht leer
++	Satz markiert
	(unabhängig vom Feldinhalt)

Die ersten vier Einzelbedingungen beachten auch den Typ eines Feldes (wie er bei der Feldstruktur eingegeben werden kann und beim Sortieren beachtet wird). So werden z.B. bei der Gleichheit von Zahlen alle nicht-numerischen Zeichen ignoriert (s. Sortieren).

Die drei Bedingungen mit Stern können auch miteinander verknüpft werden. Die Einzelbedingungen müssen dann alle zutreffen, damit der Satz ausgewählt wird. So bedeutet das Muster "M1*M2*M3*M4", daß das Feld mit M1 beginnen und mit M4 enden muß. Außerdem muß es M2 und M3 enthalten, jedoch nicht unbedingt in der angegebenen Reihenfolge.

Einzelbedingungen können durch Voranstellen von "--" verneint werden. Einzelbedingungen für verschiedene Felder werden mit UND verknüpft.

Es gibt zwei Arten der ODER-Verknüpfung: die lokale und die globale. Die lokale ODER-Verknüpfung wird durch ein Komma zwischen Einzelbedingungen realisiert. Sie hat eine höhere

Priorität als das UND zwischen verschiedenen Feldern. So hat folgendes Suchmuster

```
Feld1    Bed1, Bed2
Feld2    Bed3
```

die Bedeutung

```
( Bed1 (Feld1) ODER Bed2 (Feld2) ) UND Bed3 (Feld3)
```

Die globale ODER-Verknüpfung wird durch ein Semikolon repräsentiert. Alle Einzelbedingungen nach dem n-ten Semikolon aller Zeilen werden zu einer Alternative zusammengefaßt. Damit hat das Suchmuster

```
Feld1    Bed1; Bed2
Feld2    Bed3
```

die Bedeutung

```
( Bed1 (Feld1) UND Bed3 (Feld2) ) ODER Bed2 (Feld1)
```

Damit ergibt sich für die Priorität der einzelnen Konstruktionen folgende Reihenfolge:

höchste	Einzelbedingung
	Verkettung von Einzelbedingungen (UND)
	Verneinung
	lokales ODER
	UND zwischen Feldern
niedrigste	globales ODER

Wenn für das erste Feld einer Datei eine Gleich-Bedingung angegeben wurde und keine globale Alternative vorhanden ist, kann der Suchvorgang wegen der Dateistruktur optimiert werden, indem nur Sätze untersucht werden müssen, die im ersten Feld den gesuchten Text enthalten.

19
Drucken und Druckmuster

Durch die Funktion

- nach Muster drucken

wird die aktuelle Datei nach den Angaben in einem Druckmuster gedruckt. Der Name des Druckmusters wird erfragt. Werden bei der Dateiauswahl mehrere Dateien angekreuzt, wird der Druckvorgang mit jedem Druckmuster durchgeführt.

Das Druckmuster ist eine Textdatei und kann mit der Funktion

- Textdatei erstellen/aendern

editiert werden. Dazu wird ein Editorfenster in der unteren Schirmhälfte geöffnet.

19.1 DRUCKMUSTERSYNTAX

Ein Druckmuster ist eine Beschreibung für die Form, in der die Inhalte einer EUDAS-Datei ausgedruckt werden sollen. Der syntaktische Aufbau des Druckmusters ist zeilenorientiert und gehorcht folgender Syntax:

```
Druckmuster :
  [ Initialisierungsteil ]
  [ Vorspann ]
  [ Wiederholungsteil ]
  [ Nachspann ]

Initialisierungsteil :
  ( Kommandozeile | Textzeile )*
  ( GRUPPE-Anweisung | Textzeile )*

Vorspann :
  VORSPANN-Anweisung  Abschnitt

Wiederholungsteil :
  WIEDERHOLUNG-Anweisung  Abschnitt
```

```
Nachspann :
  NACHSPANN-Anweisung  Abschnitt

Abschnitt :
  Musterteil
  [ ABKUERZUNGEN-Anweisung   Abkürzungsteil ]

Musterteil :
  ( Musterzeile | Kommandozeile |
    MODUS-Anweisung | MEHR-Anweisung )*
```

Zur Notation: [] bedeutet optional, ()* beliebig häufige Wiederholung, | Alternative und keine Angabe einfache Aneinanderreihung. Die verschiedenen Zeilentypen werden weiter unten beschrieben.

Zusätzlich gilt die Bedingung, daß von Vorspann, Wiederholungsteil und Nachspann mindestens einer vorhanden sein muß.

Im Druckmuster gibt es 6 verschiedene Zeilentypen:

Kommandozeilen

Eine Kommandozeile beginnt mit "%%" in der ersten und zweiten Spalte. Der Inhalt der Zeile ab Spalte 3 wird dem ELAN-Compiler übergeben. Die Bedeutung dieser Zeilen ergibt sich aus dem in 19.4 beschriebenen Übersetzungsmechanismus.

Anweisungen

Anweisungen beginnen mit "%" in der ersten Spalte und dienen zur Steuerung des Druckgenerators. Der Name der Anweisung muß in Großbuchstaben und ohne Leerzeichen geschrieben werden. Davor dürfen sich noch Leerzeichen befinden. Anschließend können noch Parameter folgen, die nur durch Leerzeichen getrennt aneinander gereiht werden. Die Syntax einer Anweisung ähnelt der eines Operators in ELAN.

Textzeilen

Textzeilen sind die nicht anderweitig markierten Zeilen im Initialisierungsteil. Sie werden unverändert an den Anfang jeder Druckdatei gestellt.

Musterzeilen

Musterzeilen sind nicht besonders gekennzeichnete Zeilen im Musterteil. Sie enthalten Feldmuster und werden nach dem Einsetzen von Inhalten in die Ausgabedatei übernommen. Die Interpretation der Musterzeilen wird in 19.3 beschrieben.

Abkürzungszeilen

Abkürzungszeilen markieren den Beginn einer Abkürzung im Abkürzungsteil eines Abschnittes. Sie werden durch "&" in der ersten Spalte gekennzeichnet. Darauf folgt ohne Zwischenraum der Name einer Abkürzung (ohne Leerzeichen) und danach durch Leerzeichen getrennt ein Semikolon. Der Name der Abkürzung wird bei der Übersetzung durch einen Refinementnamen ersetzt und die Zeile dem ELAN-Compiler übergeben. Der Rest der Zeile kann also den Beginn eines wertelieferndn Refinements enthalten.

Programmzeilen

Programmzeilen sind die nicht markierten Zeilen im Abkürzungsteil. Sie werden unverändert an den ELAN-Compiler übergeben. Der erlaubte Inhalt richtet sich nach dem Übersetzungsmechanismus (19.4).

19.2 DER DRUCKVORGANG

Der Druckvorgang besteht im wesentlichen darin, daß für alle zu bearbeitenden Sätze der Wiederholungsteil einmal interpretiert wird und das Ergebnis in eine Ausgabedatei geschrieben wird, die dann gedruckt werden kann. Wenn mindestens ein Satz markiert ist, werden alle markierten Sätze der virtuellen Datei bearbeitet, ansonsten alle durch die Suchbedingung erfaßten.

Eine **Gruppe** ist eine Folge von Sätzen, die in einem bestimmten Merkmal übereinstimmen. Durch eine GRUPPE-Anweisung der Form

```
% GRUPPE n  Ausdruck
```

werden aufeinanderfolgende Sätze mit gleichem Wert des angegebenen Ausdrucks gruppiert. Über die Nummer 'n' kann festgestellt werden, ob sich das angegebene Merkmal verändert hat.

Dies geschieht mit der Prozedur

```
BOOL PROC gruppenwechsel (INT CONST gruppennr)
```

Immer wenn zwischen zwei Sätzen ein Gruppenwechsel stattfindet, wird beim vorigen Satz der Nachspann und beim folgenden Satz der Vorspann einmal interpretiert. Dies führt dazu, daß entsprechende Vorspann- bzw. Nachspannzeilen gedruckt werden.

Vor dem ersten und nach dem letzten zu bearbeitenden Satz wechseln alle Gruppen, d.h. dort wird immer ein Vorspann bzw. Nachspann erzeugt.

Ist ein zu interpretierender Abschnitt nicht vorhanden, so wird an dieser Stelle keine Ausgabe erzeugt. Die Textzeilen des Initalisierungsteils werden auf jeden Fall bei Beginn des Druckvorganges in die Ausgabedatei geschrieben. Falls die Ausgabedatei voll ist, wird eine neue Datei angefangen und diese Zeilen erneut an den Anfang gestellt.

Satz-nummer	Gruppen-merkmal	Ausgabe	
			Initialisierungsteil
1	x		Vorspann
		WDH-Teil	
2	x	WDH-Teil	
			Nachspann
3	y		Vorspann
		WDH-Teil	
4	y	WDH-Teil	
5	y	WDH-Teil	
			Nachspann
ENDE			

Ablauf des Druckvorganges mit Gruppen

Normalerweise werden die Ausgaben der einzelnen Abschnitte hintereinander in der Ausgabedatei plaziert. Durch Angabe einer Nummer als Parameter in der WIEDERHOLUNG-Anweisung können auch soviel Spalten wie angegeben nebeneinander gedruckt werden. Die Spaltenbreite wird dabei durch das Dateilimit ('maxlinelength') festgelegt. Alternativ kann die Spaltenbreite auch als zweiter Parameter angegeben werden.

Vorspann und Nachspann werden jedoch auf jeden Fall wieder in eine eigene Zeile geschrieben, der Spaltendruck also unterbrochen.

19.3 INTERPRETATION VON MUSTERZEILEN

Musterzeilen können Feldmuster enthalten, die bei der Interpretation durch entsprechende Inhalte ersetzt werden, ehe die Musterzeile in die Ausgabedatei übernommen wird. Der Beginn eines Feldmusters wird durch ein Musterzeichen ("&" oder "%") markiert. Wo und wie der Inhalt eingesetzt wird, kann durch folgende Variationen angegeben werden:

Typ	Beispiel	Position	Länge	bündig
1	&Name	fest	variabel	links
2	%Name	variabel	variabel	links
3	&Name&&&	fest	fest	links
4	%Name%%%	variabel	fest	links
5	&&&Name&	fest	fest	rechts
6	%%%Name%	variabel	fest	rechts

Der in dem Feldmuster angegebene Name muß Name einer Abkürzung in irgendeinem Abkürzungsteil oder eines Feldes sein. Der Name darf kein Leerzeichen oder Musterzeichen enthalten. Falls dies doch der Fall ist, muß der Name in spitze Klammern eingeschlossen werden.

Bei fester Länge wird bei zu kurzem Inhalt mit Leerzeichen aufgefüllt, bei zu langem Inhalt abgeschnitten. Bei linksbündigem Einsetzen geschieht dies an der rechten, sonst an der linken Seite.

Feldmuster variabler Länge können je nach Inhalt dazu führen, daß der folgende Teil der Musterzeile verschoben wird. Für diesen Einsetzprozeß gelten die folgenden Regeln:

- Position

  Feldmuster fester Position (mit "&" beginnend) werden immer in der Position eingesetzt, in der sie stehen. Feldmuster variabler Position (mit "%" beginnen) können nach rechts verschoben werden, wenn vorherige Inhalte länger als ihre Muster sind, und nach links, wenn Modus 1 oder 3 eingestellt ist und vorherige Inhalte kürzer sind.

- Länge

  Feldmuster variabler Länge erhalten auf jeden Fall den Platz, der durch die Länge des Feldmusters reserviert ist. Sind die Inhalte kürzer, kann der gewonnene Platz als Reserve für andere Feldmuster verwendet werden; sind die Inhalte länger, so wird der Inhalt so weit eingesetzt, wie noch Reserve vorhanden ist und der Rest abgeschnitten.

  Muß in ein Feldmuster variabler Länge ein leerer Inhalt eingesetzt werden, so werden beim Drucken auch die auf das Feldmuster folgenden Leerzeichen unterdrückt, falls vor dem Feldmuster noch ein Leerzeichen steht.

  Feldmuster fester Länge werden immer in ihrer reservierten Länge eingesetzt. Sie werden im folgenden behandelt wie Feldmuster variabler Länge, deren Inhalt so lang ist wie das Feldmuster.

- Verteilung

  Die Verteilung der verschiebbaren Feldmuster auf der Zeile geschieht jeweils in dem Abschnitt zwischen zwei Feldmustern fester Position bzw. Zeilenanfang oder Zeilenende. Für jeden Abschnitt wird festgestellt, wieviel Stellen die Inhalte insgesamt mehr oder weniger als ihre Muster benötigen.

  Der Längenausgleich geschieht zwischen dem letzten Feldmuster und dem Ende des Abschnitts. Dort wird ein Pufferplatz bestimmt, der bei Überlänge bis auf ein Leerzeichen verkleinert werden kann und an dem bei Unterlänge zusätzliche Leerzeichen eingefügt werden.

  Außer am Pufferplatz wird an keinem Mustertext des Abschnitts etwas geändert. Zwischentexte zwischen den Feldmustern werden unverändert übertragen und mit den umgebenden Feldmustern verschoben.

  Als Pufferplatz wird die erste Lücke vom Abschnittsende her gesehen verwendet, die mehr als ein Leerzeichen enthält. Ist keine solche Lücke vorhanden, wird das Ende des Abschnitts verwendet, falls dort ein Leerzeichen steht, und sonst das Ende des letzten Feldmusters.

  Die durch den Pufferplatz und kürzere Inhalte gewonnene Reserve wird von links an die Feldmuster mit Überlänge verteilt, bis die Reserve verbraucht ist.

- Zeilenende

  Das Zeilenende wird als ein Quasi-Feldmuster mit fester Position aufgefaßt, das am Limit der Druckmusterdatei steht. Es sind also keine Einsetzungen möglich, die über das Limit der Datei hinausgehen. Als Pufferplatz wird hier jedoch die erste Blanklücke vom Zeilenende her verwendet, damit Mustertexte am Zeilenende gegebenenfalls stehenbleiben. Ist keine solche Lücke vorhanden, so wird das Zeilenende als Pufferplatz verwendet.

  Obwohl nicht als Pufferplatz ausgewiesen, kann der Raum zwischen Zeilenende und Dateilimit als Reserve verwendet werden.

Der Einsetzmechanismus kann durch die MODUS-Anweisung mit einem Parameter verändert werden. Folgende Modi stehen zur Verfügung:

Modus	Effekt
1	Normalmodus. "%"-Feldmuster werden auch nach links geschoben. Keine Zeilenwiederholung.
2	Tabellenmodus. "%"-Feldmuster werden nicht nach links geschoben. Keine Zeilenwiederholung.
3	Normalmodus mit Zeilenwiederholung. "%"-Feldmuster werden auch nach links geschoben. Zeilenwiederholung ohne Zwischentexte.
4	Tabellenmodus mit Zeilenwiederholung. "%"-Feldmuster werden nicht nach links geschoben. Zeilenwiederholung mit Zwischentexten.

Bei Zeilenwiederholung werden Inhalte in einer folgenden Zeile fortgesetzt, falls sie in der ersten Zeile nicht untergebracht werden konnten. Dazu wird die Musterzeile mit den Restinhalten erneut interpretiert. Je nach Modus werden auch die Zwischentexte noch wiederholt. Der Restinhalt umfaßt immer noch das ganze letzte Wort, das nicht mehr auf die vorige Zeile paßte. Es findet also ein Umbruch statt. Die Positio-

nen, die in der vorigen Zeile vom Anfang des Wortes eingenommen würden, werden durch Leerzeichen ersetzt.

Durch die MEHR-Anweisung mit einem Parameter kann die Zahl der Zeilenwiederholungen für die nächste Musterzeile festgesetzt werden. Dies hat jedoch nur eine Auswirkung, falls Zeilenwiederholung zugelassen ist. Stehen zur Interpretation keine Restinhalte mehr zur Verfügung, wird mit leeren Inhalten weitergearbeitet. Kann ein Inhalt bei der vorgegebenen Anzahl von Zeilen nicht ganz dargestellt werden, wird der Rest nicht ausgegeben.

19.4 ANSCHLUSS ZUM ELAN-COMPILER

Falls in einem Druckmuster Abkürzungen oder Kommandozeilen vorkommen, wird das Druckmuster in ein ELAN-Programm umgewandelt, das dann vom ELAN-Compiler übersetzt wird.

Alle Zeilen eines Abkürzungsteils werden direkt in das Programm übernommen, wobei der Name einer Abkürzung durch einen beliebig gewählten Refinementnamen ersetzt wird ("abk" + eine laufende Nummer). Alle Abkürzungen und Refinements werden als globale Refinements definiert, also außerhalb von Prozeduren. Dadurch wird erreicht, daß sie an jeder Stelle verwendet werden können.

Damit eine Abkürzung richtig als Refinement übersetzt wird, muß sie ein TEXT-Objekt als Wert liefern. Die anderen Refinements sind beliebig, da sie nur in selbstdefinierten Anweisungen verwendet werden. Die Refinements der Abkürzungen werden in einer Zuweisung an eine TEXT-Variable verwendet, damit der Druckgenerator auf den entsprechenden Wert zugreifen kann.

Jeder Abschnitt wird dagegen als eine Prozedur übersetzt. Eine Folge von Musterzeilen wird in eine Anweisung übersetzt, diese Musterzeilen einzusetzen und zu drucken. Eine "%%"-Anweisung wird einfach unverändert dazwischengeschrieben. Die Prozedur für den Wiederholungsteil wird einmal für jeden ausgewählten Satz aufgerufen, die Vorspann- und die Nachspann-Prozedur einmal bei jedem Gruppenwechsel.

Anweisungen im Initialisierungsteil werden an den Anfang des Programms als globale Definitionen gestellt.

Findet sich in dem erzeugten ELAN-Programm ein Fehler, der durch den Druckgenerator nicht erkannt werden konnte (z.B. eine Abkürzung liefert keinen Wert), so muß der ELAN-Compiler diesen Fehler erkennen. Anschließend zeigt er das

erzeugte Programm zusammen mit seinen Fehlermeldungen im Paralleleditor. Sie müssen nun die Fehlermeldung lokalisieren und anhand der eben gegebenen Hinweise in das ursprüngliche Druckmuster zurückübersetzen, damit Sie dort den Fehler korrigieren können.

19.5 FEHLERMELDUNGEN

Folgende Fehlermeldungen können bei der Übersetzung eines Druckmusters auftreten:

keine schliessende Klammer in Feldmuster
  Wenn der Name in einem Feldmuster in spitze Klammern eingeschlossen werden soll, muß noch in der gleichen Zeile eine schließende Klammer vorhanden sein.

kein Kommando in Kommandozeile
  Eine Zeile, die mit "%" beginnt, enthält keinen weiteren Text.

unbekanntes Kommando
  Das erste Wort in einer Kommandozeile ist kein bekanntes Kommando.

kein % WIEDERHOLUNG gefunden
  Das Druckmuster enthält keine Anweisung, die den Beginn eines Abschnittes markiert. Es muß aber mindestens ein Abschnitt vorhanden sein.

nur GRUPPE-Anweisung erlaubt
  Im Initialisierungsteil ist nur die GRUPPE-Anweisung erlaubt.

keine ELAN-Anweisung im Initialisierungsteil nach Gruppendefinition
  Sobald im Initialisierungsteil eine GRUPPE-Anweisung aufgetreten ist, ist keine Kommandozeile mehr möglich.

illegale Gruppennummer
  In einer GRUPPE-Anweisung wurde eine zu große Nummer angegeben. Gruppen sollten von 1 an durchnumeriert werden.

diese Gruppe wurde schon definiert
  Eine Gruppennummer wurde mehrfach verwendet.

diese Abkürzung ist nicht definiert
Ein Name in einem Feldmuster tritt nicht als Feld-oder Abkürzungsname auf. Eventuell enthält er ein Leerzeichen!

dieser Abschnitt wurde schon einmal definiert
Kein Abschnitt kann mehrmals angegeben werden.

falscher Modus
In einer MODUS-Anweisung wurde ein nicht definierter Modus als Parameter angegeben.

diese Anweisung darf im Musterteil nicht vorkommen

im Abkürzungsteil darf keine Anweisung auftreten

in dieser Zeile stehen zu viele Feldmuster
Es können maximal 24 Feldmuster in einer Zeile stehen. Abhilfe: mehrere Feldmuster durch eine Abkürzung zusammenfassen.

das Druckmuster enthaelt zu viele Feldmuster
Die Gesamtanzahl der Feldmuster ist begrenzt. Abhilfe: mehrere Feldmuster durch eine Abkürzung zusammenfassen.

nach dem "&" soll direkt der Name einer Abkuerzung folgen
In einer Abkürzungszeile stehen Leerzeichen hinter dem "&".

kein Doppelpunkt nach Abkuerzung
Nach dem Abkürzungsnamen in einer Abkürzungszeile muß durch ein Leerzeichen getrennt ein Doppelpunkt folgen.

Abkuerzung mehrfach definiert
Die Abkürzung wurde unter dem gleichen Namen schon einmal, vielleicht in einem anderen Abschnitt, definiert.

das Druckmuster enthaelt zu viele Abkuerzungen
Abhilfe: mehrere Abkürzungen zu einem Ausdruck zusammenfassen.

20
Verarbeitung

20.1 FELDSTRUKTUR ÄNDERN

Mit der Funktion

- Feldstruktur aendern

können Sie

1. Neue Feldnamen anfügen
2. Feldnamen und Feldtypen ändern
3. Prüfbedingungen eingeben

Nach dem Aufruf können zunächst nach Anfrage die neuen Feldnamen im Editor angegeben werden. Die neuen Namen werden am Ende in der angegebenen Reihenfolge angefügt. Die dadurch geschaffenen neuen Felder sind zunächst leer.

Anschließend werden alle Feldnamen zur Auswahl angeboten, wobei jeweils der Feldtyp angegeben ist. Soll ein Name oder Typ geändert werden, so muß der entsprechende Name angekreuzt werden.

Die Feldtypen dienen dazu, Feldinhalte beim Sortieren und Suchen korrekt zu vergleichen. Die möglichen Typen sind:

TEXT Vergleich von Texten nach dem EUMEL-Code der einzelnen Zeichen. Dies ist Standard und sorgt für schnellstmöglichen Vergleich. Die weiteren Typen brauchen erheblich mehr Zeit.

DIN Vergleich nach DIN 5007 (s. EUMEL-Benutzerhandbuch Kapitel 11.3). Umlaute werden korrekt eingeordnet, Groß- und Kleinbuchstaben werden gleichbehandelt, Sonderzeichen werden ignoriert.

ZAHL Der Wert einer Zahl wird verglichen. Außer den Ziffern, dem Dezimalkomma und dem Minuszeichen vor der ersten Ziffer werden alle anderen Zeichen ignoriert. Das Dezimalkomma ist standardmäßig auf "," eingestellt, kann aber verändert werden (s. 21.5). Die nicht ignorierten Zeichen werden in eine REAL-Zahl umgewandelt und dann verglichen.

DATUM Es werden Daten der Form "tt.mm.jj" verglichen. In diesem Fall werden Tag und Jahr vertauscht und dann verglichen. Texte mit einer anderen Länge als 8 werden nicht verändert.

Für jedes angekreuzte Feld werden dann Name und Feldtyp zum Ändern angeboten.

Anschließend können nach Anfrage die Prüfbedingungen der Datei im Editor geändert werden. Zur Bedeutung der Prüfbedingungen s. 20.4.

20.2 SORTIEREN UND REORGANISIEREN

Eine EUDAS-Datei kann in einer beliebigen Feldreihenfolge sortiert werden. Mit dieser Angabe kann man bestimmen, welche Felder beim Vergleich zweier Sätze berücksichtigt werden sollen und in welcher Reihenfolge.

Um eine EUDAS-Datei zu sortieren, wird die Funktion

- EUDAS-Datei sortieren

verwendet. Die Sortierreihenfolge wird durch eine Feldauswahl bestimmt. Die Sortierreihenfolge wird in der Datei gespeichert und wird anschließend immer wieder verwendet, wenn keine anderen Angaben gemacht wurden.

Der Sortierzustand einer Datei wird ebenfalls gespeichert. Wenn nur wenige Sätze seit der letzten Sortierung verändert wurden, müssen auch nur diese Sätze einsortiert werden.

Wenn viele Änderungen an einer EUDAS-Datei vorgenommen worden sind, steigt ihr Platzbedarf durch viele Textleichen an. In diesem Fall empfiehlt es sich, die Datei zu reorganisieren. Auch wenn beim Sortieren viele Sätze vertauscht wurden, sollte die Datei reorganisiert werden, da beim Sortieren die physikalische Reihenfolge der Sätze nicht verändert wird. In diesem Fall ergibt sich nach dem Reorganisieren ein Geschwindigkeitsvorteil. Zum Reorganisieren wählen Sie die Funktion

- Datei reorganisieren

Diese Funktion kann gleichzeitig auch normale Textdateien reorganisieren.

20.3 KOPIEREN

Durch Kopieren kann ein Ausschnitt aus der virtuellen Datei in eine andere EUDAS-Datei kopiert werden. Es werden alle ausgewählten Sätze kopiert. Wenn mindestens ein Satz markiert ist, werden alle markierten Sätze als ausgewählt betrachtet, ansonsten alle, die durch die Suchbedingung angegeben sind. Diese Festlegung gilt auch für die folgenden Abschnitte. Die kopierten Sätze werden am Ende der Zieldatei angefügt.

Welche Felder kopiert werden sollen, wird durch das Kopiermuster angegeben. Hierbei können auch mehrere Felder zu einem verschmolzen werden. Allgemein ergeben sich die Felder der Zieldatei aus einem beliebigen ELAN-Ausdruck.

Das Kopiermuster ist ein ELAN-Programm und enthält im wesentlichen Ausdrücke der Form

```
"Feldname" K Ausdruck ;
```

Durch diese Anweisung wird der Ausdruck in das Feld der Zieldatei mit dem angegebenen Namen kopiert. Existiert dieses Feld in der Zieldatei noch nicht, so wird es als letztes angefügt. Falls die Zieldatei noch nicht existiert, wird sie eingerichtet. In diesem Fall bestimmt also die Reihenfolge der 'K'-Ausdrücke die Reihenfolge der Feldnamen in der Zieldatei.

Da die Reihenfolge der 'K'-Ausdrücke wichtig ist, dürfen diese nicht in einer IF-Anweisung stehen, sondern müssen für jeden Satz komplett in der gleichen Reihenfolge ausgeführt werden.

Die Kopierfunktion wird durch die Auswahl

- Saetze kopieren

aufgerufen.

Vor dem Kopieren wird ein Standard-Kopiermuster zum Editieren angeboten, das sich nach der Zieldatei richtet. Existiert die Zieldatei noch nicht, wird das Muster so konstruiert, daß alle Felder der virtuellen Datei unverändert kopiert werden. Wollen Sie einige Felder nicht kopieren, brauchen Sie nur die entsprechenden Zeilen zu löschen; wollen Sie die Felder in eine andere Reihenfolge bringen, müssen Sie die Zeilen umordnen.

Existiert die Zieldatei bereits, gibt das Standard-Kopiermuster an, daß alle Felder der Zieldatei einen Wert erhalten. Ist ein Feld der Zieldatei in der virtuellen Datei enthalten,

so wird dieses kopiert, ansonsten erhält das Feld einen leeren Inhalt. Sie können in diesem Fall weitere Felder angeben oder für die leeren Felder Ausdrücke formulieren.

20.4 TRAGEN

Durch Tragen werden alle ausgewählten Sätze der virtuellen Datei in eine andere Datei transportiert. Sie sind in der virtuellen Datei dann nicht mehr vorhanden. Damit bei diesem Vorgang keine Informationen verlorengehen können, muß die Zieldatei mindestens so viele Felder haben wie die virtuelle Datei. Normalerweise sollte sie in der Feldstruktur mit der virtuellen Datei übereinstimmen.

Die Funktion wird durch

- Saetze tragen

aufgerufen. Existiert die Zieldatei noch nicht, wird sie nach Anfrage eingerichtet. Die getragenen Sätze werden jeweils am Ende der Datei angefügt.

Beim Tragen können zusätzlich noch Konsistenzbedingungen überprüft werden. Die Prüfbedingungen sind in der Zieldatei gespeichert und können beim Ändern der Feldstruktur angegeben werden.

Die Prüfbedingung ist ein ELAN-Programm, das vor dem Tragen des jeweiligen Satzes ausgeführt wird. Durch drei spezielle Testprozeduren kann das Tragen des Satzes verhindert werden, wenn diese Prozeduren ein negatives Ergebnis liefern. Gleichzeitig wird eine Meldung in eine Protokolldatei geschrieben, die dann zur Identifizierung der fehlerhaften Sätze dienen kann.

Folgende Prüfprozeduren stehen zur Verfügung (siehe auch 23.3):

pruefe ("Feldname", Bedingung)
: Hiermit kann eine beliebige Bedingung (BOOL-Ausdruck in ELAN) überprüft werden.

wertemenge ("Feldname", "Wert1,Wert2,...,Wertn")
: Das Feld muß einen in der angegebenen Liste enthaltenen Werte annehmen.

feldmaske ("Feldname", "Maske")

Das Feld wird auf Übereinstimmung mit der Maske geprüft. Fünf spezielle Zeichen in der Maske können dabei auf mehrere Zeichen zutreffen:

'9' alle Ziffern
'a' alle Kleinbuchstaben, Umlaute, ß
'A' alle Großbuchstaben, Umlaute
'X' alle Zeichen
'*' Folge von beliebigen Zeichen

Der Stern sollte sparsam angewendet werden, da er verarbeitungsaufwendig ist.

eindeutige felder (n)

Die Zahl 'n' gibt an, die wieviel ersten Felder der Zieldatei eindeutig sein müssen. Stimmt der zu tragende Satz mit einem Satz der Zieldatei in diesen Feldern überein, wird eine Fehlermeldung erzeugt.

Es können auch einzelne Sätze manuell getragen werden. In diesem Fall wird die Prüfbedingung nicht getestet. Dies geschieht durch die Funktion

- Satz tragen

Als Umkehrung kann auch der letzte Satz der Zieldatei zurückgeholt und in der virtuellen Datei an der aktuellen Stelle eingefügt werden. Dazu dient die Funktion

- Satz holen

Die Zieldatei kann so als Einzelsatzspeicher dienen.

20.5 NACH VORSCHRIFT ÄNDERN

Die ausgewählten Sätze der virtuellen Datei können automatisch nach einer Verarbeitungsvorchrift geändert werden. Die Verarbeitungsvorschrift ist ein ELAN-Programm, in dem mit Hilfe des Operators 'V' Änderungen angegeben werden.

Aufgerufen wird die Funktion durch die Auswahl

- nach Vorschrift aendern

Dabei muß der Name einer Verarbeitungsvorschrift angegeben werden. Diese ist eine normale Textdatei und kann wie ein Druckmuster erstellt werden.

Der Operator 'V' wird in der Form

```
"Feldname" V TEXT-Ausdruck ;
```

geschrieben. Das angegebene Feld erhält den Inhalt, der durch den Ausdruck angegeben ist. Änderungen an Koppeldateien werden wie im Dialog behandelt (s. 17.3).

IV.

EUDAS FÜR PROGRAMMIERER

21
Struktur von EUDAS-Dateien

EUDAS-Dateien können unabhängig von EUDAS über den Datentyp EUDAT manipuliert werden. Die zur Verfügung stehenden Zugriffsoperationen sind in diesem Kapitel beschrieben.

21.1 DER DATENTYP SATZ

Der Datentyp SATZ stellt einen einzelnen EUDAS-Satz dar, der intern als TEXT realisiert ist. Ein SATZ besteht aus bis zu 256 Feldern, die jeweils einen TEXT enthalten können. Nach dem Initialisieren sind alle Felder mit "" vorbelegt. Die Felder können über Nummern von 1 bis 256 angesprochen werden.

Folgende Zugriffsprozeduren stehen zur Verfügung:

```
TYPE SATZ

OP := (SATZ VAR, SATZ CONST)

PROC satz initialisieren (SATZ VAR)
```

Jeder SATZ muß vor Benutzung initialisiert werden.

```
INT PROC felderzahl (SATZ CONST)
```

Liefert die Nummer des höchsten belegten Feldes.

```
PROC feld lesen (SATZ CONST, INT CONST feldnr,
                 TEXT VAR inhalt)
```

Liest den Inhalt des Feldes 'feldnr' in 'inhalt'.

FEHLER:

* n ist keine Feldnummer
  'n' liegt außerhalb des Bereiches 1..256.

```
PROC feld bearbeiten (SATZ CONST, INT CONST feldnr,
     PROC (TEXT CONST, INT CONST, INT CONST) bearbeite)
```

Ruft 'bearbeite' auf, wobei 'bearbeite' ein Text und zwei Positionen in diesem Text übergeben werden. Die Positionen geben das erste und das letzte Zeichen des durch

'feldnr' ausgewählten Feldes an. Ist der Anfang größer als das Ende, so ist das Feld leer.
FEHLER:
* n ist keine Feldnummer
'n' liegt außerhalb des Bereiches 1..256.

```
PROC feld aendern (SATZ VAR, INT CONST feldnr,
                   TEXT CONST inhalt)
```

Schreibt 'inhalt' in das Feld mit der Nummer 'feldnr'
FEHLER:
* n ist keine Feldnummer
'n' liegt außerhalb des Bereiches 1..256.

```
INT PROC feldindex (SATZ CONST, TEXT CONST muster)
```

Falls eines der Felder 'muster' enthält, wird die Nummer dieses Feldes geliefert, sonst 0.

21.2 DER DATENTYP EUDAT

Der Datentyp EUDAT muß ähnlich wie ein FILE an einen benannten oder unbenannten Datenraum angekoppelt werden. Der Datenraum hat anschließend den Typ 3243. Weitere Zugriffe auf eine EUDAT-Variable sind erst nach erfolgtem Ankoppeln zulässig. Anderenfalls können undefinierte Fehler entstehen.

```
TYPE EUDAT
```

```
PROC oeffne (EUDAT VAR, TEXT CONST dateiname)
```

Koppelt die EUDAT-Variable an die EUDAS-Datei mit dem Namen 'dateiname' an. Die Datei wird eingerichtet, falls sie noch nicht existiert.
FEHLER:
* Datei ist keine EUDAS-Datei
Die angegebene Datei hat nicht den Typ 3243.

```
PROC oeffne (EUDAT VAR, DATASPACE CONST ds)
```

Koppelt die EUDAT-Variable an den Datenraum 'ds'.
FEHLER:
* Datei ist keine EUDAS-Datei
Der Datenraum wurde bereits verwendet und hat nicht den Typ 3243.

Eine EUDAS-Datei ist in Felder und Sätze unterteilt. Die Felder werden beim Zugriff über Nummern angesprochen. Jedem Feld ist jedoch zur Identifikation ein TEXT als Feldname zugeordnet. Die Feldnamen werden als SATZ gespeichert, wobei jedes Feld seinen zugeordneten Namen enthält.

```
INT PROC felderzahl (EUDAT CONST)
```

Liefert Anzahl der benannten Felder. Ist zu Anfang 0.

```
PROC feldnamen aendern (EUDAT VAR,
                        SATZ CONST neue namen)
```

Setzt die Feldnamen einer Datei. Ist 'felderzahl (neue namen)' größer als die Felderzahl der Datei, so wird die Felderzahl der Datei entsprechend heraufgesetzt.

```
PROC feldnamen lesen (EUDAT CONST, SATZ VAR namen)
```

Liefert alle Feldnamen in einer SATZ-Variablen.

Eine EUDAS-Datei enthält drei zusätzliche Notiztexte. Zwei davon sind bereits reserviert, und zwar:

1: Prüfbedingungen
2: Datum der letzten Änderung

```
PROC notizen lesen (EUDAT CONST, INT CONST notiz nr,
                    TEXT VAR notizen)
```

Schreibt die Notizen der EUDAS-Datei in 'notizen'.

```
PROC notizen aendern (EUDAT VAR, INT CONST notiz nr,
                      TEXT CONST notizen)
```

Ändert die Notizen. Alte Notizen werden dabei überschrieben.

21.3 SATZPOSITION

Eine EUDAS-Datei läßt sich sequentiell vorwärts und rückwärts bearbeiten. Dazu gibt es eine aktuelle Satzposition. Ein bestimmter Satz kann auch direkt angesprungen werden. Die Prozeduren, die nach dem Inhalt des ersten Feldes suchen, arbeiten besonders schnell, da die entsprechenden Sätze über eine Hashmethode gefunden werden.

```
INT PROC satznr (EUDAT CONST)
```

Liefert aktuelle Satzposition.

```
INT PROC saetze (EUDAT CONST)
```
Liefert Anzahl der Sätze.

```
BOOL PROC dateiende (EUDAT CONST)
```
Liefert TRUE, wenn 'satznr' groesser als 'saetze' ist. Die letzte erreichbare Satzposition liegt um eins hinter dem letzten Satz (um auch am Ende anfügen zu können).

```
PROC auf satz (EUDAT VAR, INT CONST satznr)
```
Positioniert auf den gewünschten Satz. Bei nicht existierenden Sätzen wird auf den ersten bzw. hinter den letzten Satz gesprungen.

```
PROC weiter (EUDAT VAR)
```
Geht einen Satz weiter, jedoch nicht über das Dateiende hinaus.

```
PROC zurueck (EUDAT VAR)
```
Geht einen Satz zurück, falls der erste Satz noch nicht erreicht ist.

```
PROC auf satz (EUDAT VAR, TEXT CONST muster)
```
Positioniert auf den ersten Satz, der als erstes Feld 'muster' enthält, anderenfalls hinter den letzten Satz.

```
PROC weiter (EUDAT VAR, TEXT CONST muster)
```
Geht weiter, bis das erste Feld 'muster' enthält, bzw. bis hinter den letzten Satz.

```
PROC zurueck (EUDAT VAR, TEXT CONST muster)
```
Geht zurück, bis das erste Feld 'muster' enthält, bzw. auf den ersten Satz der EUDAS-Datei.

21.4 SATZZUGRIFFE

Der aktuelle Satz ist ein SATZ-Objekt. Auf die Felder des aktuellen Satzes kann direkt zugegriffen werden.

```
PROC feld lesen (EUDAT CONST, INT CONST feldnr,
                 TEXT VAR inhalt)
```
Wirkt wie 'feld lesen' auf den aktuellen Satz.

```
PROC feld aendern (EUDAT VAR, INT CONST feldnr,
                   TEXT CONST inhalt)
```

Wirkt wie 'feld aendern' auf den aktuellen Satz.

```
PROC feld bearbeiten (EUDAT CONST, INT CONST feldnr,
     PROC (TEXT CONST, INT CONST, INT CONST) bearbeite)
```

Wirkt wie 'feld bearbeiten' auf den aktuellen Satz.

Der aktuelle Satz kann auch als Ganzes bearbeitet werden.

```
PROC satz lesen (EUDAT CONST, SATZ VAR satz)
```

Liefert den aktuellen Satz.

```
PROC satz aendern (EUDAT VAR, SATZ CONST satz)
```

Ersetzt den aktuellen Satz durch 'satz'.

```
PROC satz einfuegen (EUDAT VAR, SATZ CONST satz)
```

Fügt 'satz' vor dem aktuellen Satz ein.
FEHLER:
* EUDAS-Datei voll
  Eine EUDAS-Datei faßt mindestens 5000 Sätze.

```
PROC satz loeschen (EUDAT VAR)
```

Löscht den aktuellen Satz.

21.5 SORTIEREN UND REORGANISIEREN

Zum Sortieren können für die einzelnen Felder Typen angegeben werden, damit auch Zahlen und Daten richtig sortiert werden können. Außerdem kann die Feldreihenfolge angegeben werden, nach der sortiert werden soll.

```
PROC feldinfo (EUDAT VAR, INT CONST feldnr, info)
```

Setzt den Feldtyp des Feldes 'feldnr'. Es bedeuten

- -1 : normaler Text (Standard)
- 0 : Text nach DIN. Ziffern und Sonderzeichen werden ignoriert. Groß-und Kleinbuchstaben gelten gleich. Umlaute werden beachtet.
- 1 : Zahl (beim Vergleich werden alle Zeichen außer Ziffern ignoriert).
- 2 : Datum. Es werden Daten der Form "tt.mm.jj" verglichen.

INT PROC feldinfo (EUDAT CONST, INT CONST feldnr)
Der Feldtyp des angegebenen Feldes wird geliefert. Zu Anfang ist -1 voreingestellt.

INT PROC unsortierte saetze (EUDAT CONST)
Liefert die Anzahl von Sätzen, die seit dem letzten Sortiervorgang geändert wurden. Bei einer neuen Datei, die noch nie sortiert wurde, wird immer 0 geliefert.

PROC dezimalkomma (TEXT CONST komma)
Stellt das Dezimalkomma ein, das beim Vergleich von Zahlen gelten soll.
FEHLER:
* Nicht erlaubtes Dezimalkomma
Nur Texte der Länge 1 sind zugelassen.

TEXT PROC dezimalkomma
Liefert das eingestellte Dezimalkomma (",") ist voreingestellt).

PROC sortiere (EUDAT VAR, TEXT CONST reihenfolge)
Sortiert die Datei in der von 'reihenfolge' angegebenen Reihenfolge. Dabei enthält 'reihenfolge' als i-tes Zeichen den Code der Feldnummer, die als i-te in der Sortierung berücksichtigt werden soll.

PROC sortiere (EUDAT VAR)
Sortiert die Datei in der zuletzt eingestellten Reihenfolge. Wurde noch keine Reihenfolge angegeben, wird die Datei in der Feldreihenfolge sortiert.

TEXT PROC sortierreihenfolge (EUDAT CONST)
Liefert die zuletzt eingestellte Reihenfolge. Wurde noch nicht sortiert, so wird "" geliefert.

Nach umfangreichen Änderungen an einer EUDAS-Datei ist eine Reorganisation sinnvoll, um "Textleichen" zu beseitigen.

PROC reorganisiere (TEXT CONST dateiname)
Die EUDAS-Datei mit dem Namen 'dateiname' wird reorganisiert.

22
Verwaltung der offenen Dateien

Die in diesem Kapitel beschriebene Schnittstelle verbindet mehrere EUDAS-Dateien zu einem großen Dateimodell. Diese Schnittstelle ist in vielen Teilen für die interne EUDAS-Anwendung ausgelegt. Bei einigen Prozeduren werden aus Effizienzgründen keinerlei Überprüfungen auf illegale Aufrufe oder Parameter durchgeführt. Wollen Sie eine solche Prozedur dennoch verwenden, sollten Sie die Einhaltung der angegebenen Bedingungen sorgfältig überprüfen.

22.1 DATEIVERWALTUNG

Mit 'oeffne' wird eine Datei zum Bearbeiten geöffnet. Mit 'kette' und 'kopple' können weitere Dateien dazugekettet bzw. dazugekoppelt werden. Durch 'sichere' können veränderte Kopien zurückgeschrieben werden. Durch 'dateien loeschen' werden die internen Kopien gelöscht.

Mit 'anzahl dateien' kann die Anzahl der vorhandenen Dateien erfragt werden. 'anzahl koppeldateien' gibt Auskunft darüber, wieviel Dateien davon gekoppelt sind. 'aendern erlaubt' gibt den Status wieder, der beim Öffnen der ersten Datei angegeben wurde. 'inhalt veraendert' gibt an, ob die angegebene Datei verändert wurde. Mit 'eudas dateiname' können die Namen der geöffneten Dateien erfragt werden. Bei jedem 'oeffne' wird 'dateiversion' um 1 erhöht. Dies dient dazu, ein erfolgtes neues Öffnen von anderen Stellen aus zu entdecken.

```
PROC oeffne (TEXT CONST dateiname,
             BOOL CONST aendern erlaubt)
```

Falls Ändern erlaubt sein soll, wird eine Kopie der angegebenen Datei zur Bearbeitung für EUDAS angelegt. Vorher geöffnete Dateien werden gelöscht. Die Änderungserlaubnis wird entsprechend gesetzt. Es wird die Satzposition der EUDAS-Datei angenommen (Ausnahme: steht die EUDAS-Datei hinter dem letzten Satz, wird auf Satz 1 positioniert). 'dateiversion' sowie 'anzahl dateien' werden um 1 erhöht.

FEHLER:

* Datei nicht gesichert
  Eine vorher geöffnete Datei war verändert, aber nicht gesichert.
* Datei existiert nicht
  Die angegebene Datei ist nicht vorhanden.
* Datei ist keine EUDAS-Datei
  Die angegebene Datei hat nicht den für EUDAS-Dateien festgelegten Typ.

PROC kopple (TEXT CONST dateiname)

Die angegebene Datei wird zu den bereits geöffneten Dateien dazugekoppelt. Falls Ändern erlaubt ist, wird eine Kopie dieser Datei verwendet. Dabei werden die ersten Felder der Datei, die bereits in der Hauptdatei vorhanden sind, als Koppelfelder festgelegt. Alle weiteren Felder werden zusätzlich zu den bisherigen angelegt. 'anzahl dateien' und 'anzahl koppeldateien' werden um 1 erhöht.

FEHLER:

* keine Datei geoeffnet
  Es muß vorher eine Datei geöffnet werden.
* Zuviel Dateien geoeffnet
  Die Anzahl der gleichzeitig geöffneten Dateien ist begrenzt.
* Datei existiert nicht
  Die angegebene Datei ist nicht vorhanden.
* Datei ist keine EUDAS-Datei
  Die angegebene Datei hat nicht den für EUDAS-Dateien festgelegten Typ.
* Zu viele Felder
  Die Anzahl der Felder insgesamt ist begrenzt.
* Zu viele Koppelfelder
  Die Anzahl der Koppelfelder ist begrenzt.
* keine Koppelfelder vorhanden
  Das erste Feld der zu koppelnden Datei ist in der Hauptdatei nicht vorhanden (unterschiedliche Feldnamen).

PROC kette (TEXT CONST dateiname)

Die angegebene Datei wird an die Hauptdatei angekettet, d.h. die Sätze der neuen Datei werden am bisherigen Dateiende angefügt. Falls Ändern erlaubt ist, wird eine Kopie dieser Datei verwendet. Die zu kettende Datei muß in der

Feldstruktur nicht mit der Hauptdatei übereinstimmen. Die aktuelle Satzposition wird beibehalten. 'anzahl dateien' wird um 1 erhöht.
FEHLER:
* keine Datei geoeffnet
  Es muß vorher eine Datei geöffnet werden.
* Zuviel Dateien geoeffnet
  Die Anzahl der gleichzeitig geöffneten Dateien ist begrenzt.
* Datei existiert nicht
  Die angegebene Datei ist nicht vorhanden.
* Datei ist keine EUDAS-Datei
  Die angegebene Datei hat nicht den für EUDAS-Dateien festgelegten Typ.

PROC sichere (INT CONST dateinr, TEXT CONST dateiname)
Die geöffneten Dateien werden in der Reihenfolge ihres Öffnens durchnumeriert (von 1 an). Die Datei mit der angegebenen Nummer wird unter dem angegebenen Namen gesichert, aber selbst nicht verändert. Die vorher unter diesem Namen vorhandene Datei wird gelöscht. War die zu sichernde Datei verändert worden, so wird sie anschließend als nicht verändert angesehen.
Bedingungen:
1 <= dateinr <= anzahl dateien

PROC dateien loeschen (BOOL CONST auch geaenderte)
Es werden alle geöffneten Dateikopien gelöscht. EUDAS wird wieder in den Anfangszustand versetzt. Wird 'auch geaenderte' angegeben, wird bei geänderten, aber nicht gesicherten Dateien die Fehlermeldung unterdrückt.
FEHLER:
* Datei nicht gesichert
  Eine vorher geöffnete Datei war verändert, aber nicht gesichert.

INT PROC anzahl dateien
Gibt die Anzahl der insgesamt geöffneten Dateien an.

INT PROC anzahl koppeldateien
Gibt die Anzahl der gekoppelten Dateien an.

```
BOOL PROC aendern erlaubt
```

Reflektiert den Status, der bei 'oeffne' gesetzt wurde.

```
BOOL PROC inhalt veraendert (INT CONST dateinr)
```

Gibt an, ob die geöffnete Datei mit der angegebenen Nummer verändert wurde. Wird ggf von 'sichere' zurückgesetzt.

Bedingung:

1 <= dateinr <= anzahl dateien

```
TEXT PROC eudas dateiname (INT CONST dateinr)
```

Liefert den Namen, unter dem die entsprechende Datei geöffnet wurde.

Bedingung:

1 <= dateinr <= anzahl dateien

```
INT PROC dateiversion
```

Wird bei jedem 'oeffne' zyklisch erhöht.

22.2 FELDSTRUKTUR

Die einzelnen Sätze der kombinierten EUDAS-Datei sind in Felder unterteilt. Diese setzen sich zusammen aus den Feldern der Hauptdatei und der einzelnen Koppeldateien, wobei die Koppelfelder jeweils nur einmal auftauchen.

'anzahl felder' liefert die Anzahl der vorhanden Felder. Mit 'feldnamen lesen' und 'feldnamen bearbeiten' können die Feldnamen abgefragt werden. 'feldnummer' liefert einen Index für einen vorgegebenen Feldnamen, da die Felder immer über Nummern angesprochen werden.

Die Prozeduren 'feld lesen' und 'feld bearbeiten' ermöglichen den Zugriff auf den Feldinhalt des aktuellen Satzes; durch 'feld aendern' kann dieser Inhalt abgeändert werden.

```
INT PROC anzahl felder
```

Liefert die Anzahl der vorhanden Felder.

```
PROC feldnamen lesen (INT CONST feldnr,
                      TEXT VAR feldname)
```

Liefert in 'feldname' den Namen des Feldes mit der Nummer 'feldnr'.

Bedingung:

1 <= feldnr <= anzahl felder

```
PROC feldnamen bearbeiten (INT CONST feldnr,
     PROC (TEXT CONST, INT CONST, INT CONST) bearbeite)
```

Die Prozedur 'bearbeite' wird aufgerufen. Als Parameter werden ein Text und Anfangs- und Endposition des gewünschten Feldnamens in diesem Text übergeben. Verhindert unnötiges Kopieren des Feldnamens in eine TEXT-Variable. Der übergebene Text darf nicht verändert werden !

Bedingung:

```
  1 <= feldnr <= anzahl felder
```

```
INT PROC feldnummer (TEXT CONST feldname)
```

Liefert den index zu dem angegebenen Feldnamen. Falls ein solcher Name nicht existiert, wird 0 geliefert.

```
PROC feld lesen (INT CONST feldnr, TEXT VAR inhalt)
```

Liefert den Inhalt des angegebenen Feldes.

Bedingung:

```
  1 <= feldnr <= anzahl felder
```

```
PROC feld bearbeiten (INT CONST feldnr,
     PROC (TEXT CONST, INT CONST, INT CONST) bearbeite)
```

Die Prozedur 'bearbeite' wird aufgerufen. Der Feldinhalt des angegebenen Feldes steht im übergebenen Text innerhalb der Grenzen. Ist die Obergrenze kleiner als die Untergrenze, so ist das Feld leer.

Bedingung:

```
  1 <= feldnr <= anzahl felder
```

```
PROC feld aendern (INT CONST feldnr, TEXT CONST inhalt)
```

Ändert den Inhalt des angegebenen Feldes.

Bedingung:

```
  NOT ende der datei
  1 <= feldnr <= anzahl felder
```

22.3 POSITIONIERUNG

Das virtuelle Dateimodell von EUDAS verfügt ebenfalls über eine Satzposition, die verändert werden kann.

Durch 'satznummer' wird die aktuelle Satznummer geliefert, beim Koppeln kann über 'satzkombination' die Reihenfolge der Koppelkombinationen bestimmt werden. 'dateiende' zeigt an, ob die Satzposition hinter dem letzten Satz liegt. Mit

'weiter' und 'zurueck' erfolgt die eigentliche Positionierung. Hier kann außer der Positionierung um Einzelsätze auch die Positionierung auf den nächsten ausgewählten oder markierten Satz angefordert werden. Mit 'auf satz' schließlich kann ein bestimmter Satz angesprungen werden.

INT PROC satznummer

Liefert die Nummer des aktuellen Satzes. Die Sätze werden von 1 an durchnumeriert, wobei über die geketteten Dateien weitergezählt wird.
Bedingung:
anzahl dateien > 0

INT PROC satzkombination

Liefert die laufende Nummer der Koppelkombination des aktuellen Satzes. Wird nur durch 'weiter' im Einzelsatzmodus erhöht. Normalerweise 1.
Bedingung:
anzahl dateien > 0

BOOL PROC dateiende

Gibt an, ob die Satzposition hinter dem letzten Satz liegt.

PROC weiter (INT CONST modus)

Erhöht die aktuelle Satzposition. Für 'modus' gibt es 3 Möglichkeiten:

1: Falls eine weitere Satzkombination besteht, wird diese eingenommen, sonst zum nächsten Satz.
2: Zum nächsten durch Suchbedingung ausgewählten Satz. Wird optimiert.
3: Zum nächsten markierten Satz. Wird optimiert.

Ist kein Satz mehr vorhanden, wird die Satzposition hinter dem letzten Satz eingenommen.
Bedingung:
anzahl dateien > 0

PROC zurueck (INT CONST modus)

Geht um einen Satz zurück. Die Modusangabe ist wie bei 'weiter', jedoch wird im Modus 1 keine weitere Satzkombination ausprobiert. Die Positionierung endet bei Satz 1.
Bedingung:
anzahl dateien > 0

PROC auf satz (INT CONST satznr)

Geht auf den angegebenen Satz. Ist 'satznr' < 1, wird auf Satz 1 positioniert, ist der angegebene Satz nicht vorhanden, wird hinter den letzten Satz positioniert. Es wird jeweils die erste Satzkombination eingenommen.

Bedingung:

anzahl dateien > 0

22.4 ÄNDERUNGEN

Sätze des Dateimodells können eingefügt oder gelöscht werden. Durch das Einfügen entsteht ein leerer Satz vor dem aktuellen Satz; alle weiteren Sätze rücken eine Stelle weiter. Beim Löschen wird dieser Vorgang wieder rückgängig gemacht.

Durch 'satz einfuegen' wird ein Leersatz eingefügt; durch 'satz loeschen' wird der aktuelle Satz gelöscht.

Sätze in gekoppelten Dateien werden grundsätzlich nicht gelöscht; auch beim Einfügen entsteht nicht automatisch ein Leersatz in den gekoppelten Dateien. Änderungen in den Koppeldateien (durch 'feld aendern') werden gepuffert. Durch 'aenderungen eintragen' werden die Änderungen dann in die Koppeldateien eingetragen. Dabei kann auch ein neuer Satz in die Koppeldatei eingefügt werden. Bei Positionierungen wird diese Prozedur automatisch aufgerufen.

PROC satz einfuegen

Fügt vor dem aktuellen Satz einen Leersatz ein.

Bedingung:

anzahl dateien > 0

PROC satz loeschen

Löscht den aktuellen Satz. Hat hinter dem letzten Satz keine Wirkung.

Bedingung:

anzahl dateien > 0

PROC aenderungen eintragen

Trägt die gepufferten Änderungen in die Koppeldateien ein. Dabei können die folgenden Fälle auftreten:

1. Der Satz in der Koppeldatei wird geändert.
   Dies geschieht dann, wenn vorher ein passender Satz in der Koppeldatei vorhanden war und die Koppelfelder nicht verändert wurden.
2. In der Koppeldatei wird ein neuer Satz eingefügt.
   Wenn die Koppelfelder und noch andere Felder einer Datei geändert wurden, wird in dieser Datei ein neuer Satz eingefügt.
3. Es wird neu gekoppelt.
   Wurden nur die Koppelfelder einer Datei geändert, wird ein neuer, zu diesen Feldern passender Satz gesucht. Nach 'aenderungen eintragen' erscheinen unter den Feldern der Datei die neuen Inhalte.

22.5 SUCHBEDINGUNGEN

Über 'suchbedingung' kann eine Suchbedingung eingetragen werden, die für jeden Satz geprüft werden soll. Mit 'satz ausgewaehlt' wird erfragt, ob der aktuelle Satz die Suchbedingung erfüllt. Die Suchbedingung kann mit 'suchbedingung loeschen' wieder ausgeschaltet werden.

Einzelne Sätze können auch markiert werden. Nach einem Öffnen ist zunächst kein Satz markiert. Durch 'markierung aendern' kann die Markierung eines Satzes geändert werden. 'satz markiert' fragt ab, ob der aktuelle Satz markiert ist. 'markierte saetze' liefert die Anzahl der markierten Sätze. Mit 'markierungen loeschen' werden alle Markierungen entfernt.

```
PROC suchbedingung (INT CONST feldnr,
                    TEXT CONST bedingung)
```

Stellt für das angegebene Feld die im Text als Muster angegebene Suchbedingung ein. Weitere Aufrufe verknüpfen die Bedingungen mit UND (auch wenn das gleiche Feld erneut angegeben wird).

Bedingung:

anzahl dateien > 0

1 <= feldnr <= anzahl felder

FEHLER:

* Suchmuster zu umfangreich
  Es wurden zu viele Vergleiche gefordert.

BOOL PROC satz ausgewaehlt

Gibt an, ob die Suchbedingung auf den aktuellen Satz zutrifft. Hinter dem letzten Satz wird immer FALSE geliefert.

Bedingung:

anzahl dateien > 0

PROC suchbedingung loeschen

Löscht die eingestellte Suchbedingung wieder. Anschließend sind alle Sätze ausgewählt.

Bedingung:

anzahl dateien > 0

PROC markierung aendern

Ändert die Markierung des aktuellen Satzes ins Gegenteil.

Bedingung:

anzahl dateien > 0

BOOL PROC satz markiert

Gibt an, ob der aktuelle Satz markiert ist.

Bedingung:

anzahl dateien > 0

INT PROC markierte saetze

Gibt an, wieviel Sätze insgesamt markiert sind.

Bedingung:

anzahl dateien > 0

PROC markierungen loeschen

Löscht alle Markierungen.

Bedingung:

anzahl dateien > 0

23
Verarbeitungsfunktionen

Die Verarbeitungsfunktionen arbeiten jeweils auf der aktuell geöffneten Datei. Falls mindestens ein Satz markiert ist, werden nur markierte Sätze bearbeitet, anderenfalls die durch die Suchbedingung ausgewählten Sätze.

23.1 DRUCKEN

Zum Drucken wird ein Druckmuster als Textdatei benötigt. Dessen Name muß beim Aufruf der Prozedur 'drucke' angegeben werden. Werden beim Übersetzen des Druckmusters Fehler entdeckt, so wird der Paralleleditor aufgerufen und kein Druckvorgang durchgeführt.

Normalerweise sendet der Druckgenerator die Ausgabe direkt zum Drucker. Alternativ kann die Ausgabe auch in eine Datei geschrieben werden. Dieses Verfahren kann mit 'direkt drucken' umgestellt werden. Der Aufruf

```
direkt drucken (TRUE)
```

sendet alle Dateien direkt zum Drucker, mit

```
direkt drucken (FALSE)
```

wird die Ausgabe in Dateien abgelegt. Diese Dateien erhalten Namen der Form

```
"EUDAS-Ausgabe.n"
```

wobei 'n' eine laufende Nummer zur Unterscheidung ist.

Soll die Druckausgabe in eine ganz bestimmte Datei geleitet werden, so kann vor dem Aufruf von 'drucke' die Prozedur 'druckdatei' aufgerufen werden, die als Parameter den Namen der Ausgabedatei erhält. Existiert die Datei noch nicht, wird sie eingerichtet, ansonsten wird die Ausgabe am Ende angehängt.

Die Einstellung der Ausgabedatei gilt nur für einen Druckvorgang und überschreibt für diesen Druckvorgang

'direkt drucken'. Beim nächsten Druckvorgang wird wieder die durch 'direkt drucken' festgelegte Einstellung verwendet.

Wenn beim Drucken ein großes Ausgabevolumen anfällt, kann es sinnvoll sein, die Ausgabe in mehrere kleine Dateien aufzuteilen. Dies gilt auch, wenn direkt gedruckt werden soll, da auch in diesem Fall eine Zwischendatei erzeugt werden muß. Die maximale Anzahl von Zeilen pro Datei wird durch 'max druckzeilen' angegeben.

Der dort angegeben Wert gilt nur ungefähr - ein Wechsel der Ausgabedatei findet dann statt, wenn die Ausgabedatei nach Bearbeitung eines Satzes die Maximalanzahl überschritten hat. In die neue Datei wird anschließend zuerst der Initialisierungsteil des Druckmusters kopiert, ehe mit der Ausgabe des nächsten Satzes fortgefahren wird.

Die Prozeduren im einzelnen:

PROC drucke (TEXT CONST druckmuster)

Die aktuell geöffnete Datei wird nach dem angegebenen Druckmuster gedruckt.

FEHLER:

* Datei "druckmuster" existiert nicht
  Das angegebene Druckmuster ist nicht vorhanden.
* keine Datei geoeffnet
  Zum Drucken muß eine Datei geöffnet sein.
* direkt Drucken nicht möglich
  Es ist kein Druckprogramm installiert oder der Spooler läßt sich mit 'print' nicht ansprechen. Der Druck wird abgebrochen, die Ausgabedatei ist noch vorhanden.

PROC direkt drucken (BOOL CONST ja)

Gibt an, ob die Druckausgaben direkt gedruckt oder in einer Datei gesammelt werden sollen.

PROC druckdatei (TEXT CONST ausgabedatei)

Leitet die Druckausgabe des nächsten Druckvorgangs in die Datei "ausgabedatei". Die Einstellung von 'direkt drucken' wird für diesen Druckvorgang überschrieben. Die Ausgabe wird am Ende der Datei angehängt, falls nötig, wird die Ausgabedatei vorher eingerichtet.

PROC maxdruckzeilen (INT CONST zeilen)

Stellt die maximale Anzahl von Zeilen für die Ausgabedatei ein. Beim Überschreiten dieses Wertes wird eine neue Datei angefangen. Standardwert ist 4000.

23.2 KOPIEREN

Zum selektiven Kopieren von EUDAS-Dateien wird ein Kopiermuster benötigt. Diese gibt die Zuordnung zwischen Feldern der Ziel- und der Quelldatei an. Die Quelldatei ist immer die aktuell geöffnete Datei.

Die Kopierfunktion wird durch 'kopiere' aufgerufen. Parameter sind der Name der Zieldatei und das Kopiermuster als FILE. Alternativ kann statt des Kopiermusters eine Prozedur übergeben werden, die die Kopieranweisungen erhält.

Der eigentliche Kopiervorgang wird durch den Operator "K" bewirkt. Dieser erhält den Zielfeldnamen und einen TEXT-Ausdruck als Parameter. Der Wert des TEXT-Ausdrucks wird in das jeweilige Feld der Zieldatei geschrieben.

Existiert die Zieldatei noch nicht, so wird sie mit den Feldern eingerichtet, die in den einzelnen "K"-Ausdrücken angegeben sind und zwar in der angeführten Reihenfolge. Existiert die Zieldatei, so werden gegebenenfalls noch nicht vorhandene Felder am Ende angefügt.

Die Prozedur 'std kopiermuster' liefert zu einer gegebenen Zieldatei ein Standard-Muster, das als Auswahlgrundlage dienen kann. Existiert die Zieldatei nicht, werden alle Felder der Quelldatei 1 : 1 kopiert, anderenfalls wird zu jedem Feld der Zieldatei ein passendes Feld der Quelldatei gesucht - die Feldreihenfolge richtet sich in diesem Fall nach der Zieldatei.

PROC kopiere (TEXT CONST dateiname,
FILE VAR kopiermuster)

Die aktuell geöffnete Datei wird nach den Angaben in 'kopiermuster' in die Datei 'dateiname' kopiert. Das Kopiermuster wird dem ELAN-Compiler übergeben. Tritt bei der Übersetzung ein Fehler auf, wird der Paralleleditor aufgerufen.

FEHLER:
* Datei ist keine EUDAS-Datei
  Zieldatei existiert, ist aber keine EUDAS-Datei.
* keine Datei geoeffnet
  Es muß eine virtuelle Datei vorhanden sein.

PROC kopiere (TEXT CONST dateiname, PROC kopierfunktion)
Wie oben, nur ist die Kopierfunktion gleich als Prozedur vorhanden.
FEHLER:
* Datei ist keine EUDAS-Datei
  Zieldatei existiert, ist aber keine EUDAS-Datei.
* keine Datei geoeffnet
  Es muß eine virtuelle Datei vorhanden sein.

OP K (TEXT CONST feldname, ausdruck)
Kopiert den Ausdruck in das Feld 'feldname' der Zieldatei. Dieses Feld wird eingerichtet, falls es noch nicht existiert. Dieser Operator ist nur während eines Kopiervorganges definiert (also in einem Kopiermuster oder einer Kopierfunktion). Er darf nicht in einer IF-Klausel stehen, sondern muß bei jedem Satz mit gleichem Feldnamen an der gleichen Stelle aufgerufen werden.

PROC std kopiermuster (TEXT CONST dateiname,
FILE VAR kopiermuster)
Liefert ein Standard-Kopiermuster, abhängig von der Zieldatei 'dateiname'. Existiert diese nicht, wird die Quelldatei unverändert kopiert, ansonsten richtet sich das Kopiermuster nach der Zieldatei.

23.3 TRAGEN

Durch Tragen können Sätze komplett in eine Zieldatei transportiert werden. In der Quelldatei sind sie anschließend nicht mehr vorhanden. Eine ganze Auswahl von Sätzen kann mit 'trage' transportiert werden. 'trage satz' transportiert nur den aktuellen Satz. Mit 'hole satz' kann der letzte Satz der Zieldatei wieder zurückgeholt werden, so daß eine EUDAS-Datei auch als Zwischenspeicher für Einzelsätze verwendet werden kann.

Existiert die Zieldatei bereits, muß sie mindestens so viele Felder wie die Quelldatei besitzen, damit keine Informationen verlorengehen können. Die Feldnamen müssen nicht übereinstimmen. Existiert die Zieldatei noch nicht, wird sie mit den Feldern der Quelldatei eingerichtet.

Die Tragefunktion kann um eine gleichzeitige Prüfung erweitert werden. Dabei werden Bedingungen überprüft, die bei der Zieldatei gespeichert sind. Sätze, die diese Bedingungen verletzen, werden nicht getragen. Eine entsprechende Meldung wird in eine Protokolldatei geschrieben, die als Parameter übergeben werden muß.

Die Prüfbedingungen stehen als ausführbares Programm in den Notizen der Zieldatei. Prüfbedingungen können mit mehreren Prozeduren formuliert werden. 'pruefe' nimmt eine beliebige Bedingung als Parameter und gibt bei Mißerfolg eine Meldung aus. 'wertemenge' prüft auf Übereinstimmung mit einem der angegebenen Werte. 'feldmaske' legt eine Maske für ein Feld fest, die auf den Inhalt zutreffen muß.

Mit Hilfe der Prozedur 'eindeutige felder' können Satzduplikate erkannt werden. Auch diese werden nicht getragen.

Die bei den Prüfbedingungen angegebenen Feldnamen müssen in der Quelldatei vorhanden sein. Falls eine Prüfprozedur außerhalb von 'trage' aufgerufen wird, führt eine Verletzung der Prüfbedingung zu einem 'errorstop'.

```
PROC trage (TEXT CONST dateiname,
            FILE VAR protokoll, BOOL CONST test)
```

Alle ausgewählten Sätze werden in die Datei 'dateiname' getragen. Diese wird gegebenenfalls eingerichtet. Falls 'test' angegeben ist, werden die in den Notizen der Zieldatei enthaltenen Bedingungen geprüft. Nur in diesem Fall muß 'protokoll' initialisiert sein.

FEHLER:

* kein Satz zum Tragen vorhanden
  Die Quelldatei ist leer oder es ist keine Datei geöffnet.
* Datei ist keine EUDAS-Datei
  Zieldatei existiert, ist aber keine EUDAS-Datei.
* Zieldatei hat falsche Felderzahl
  Zu wenig Felder in der Zieldatei.

PROC trage satz (TEXT CONST dateiname)

Der aktuelle Satz wird in die Datei 'dateiname' getragen.
FEHLER:

* kein Satz zum Tragen vorhanden
  Keine Datei geöffnet oder Datei ist am Ende.
* Datei ist keine EUDAS-Datei
  Zieldatei existiert, ist aber keine EUDAS-Datei.
* Zieldatei hat falsche Felderzahl
  Zu wenig Felder in der Zieldatei.

PROC pruefe (TEXT CONST feldname, BOOL CONST bedingung)

Wenn die angegebene Bedingung FALSE liefert, wird eine Meldung in die Protokolldatei geschrieben und der jeweilige Satz nicht getragen.

PROC wertemenge (TEXT CONST feldname. menge)

Es wird geprüft, ob das angegebene Feld in der Wertemenge enthalten ist. Die einzelnen Werte in der Wertemenge werden dabei durch Komma getrennt. Leerzeichen sind signifikant.

PROC feldmaske (TEXT CONST feldname, maske)

Es wird geprüft, ob das angegebene Feld zu der Maske paßt. Die Zeichen in der Maske haben dabei folgende Bedeutung:

"9"	trifft auf jede Ziffer zu
"X"	trifft auf jedes Zeichen zu
"A"	trifft auf jeden Großbuchstaben zu (einschließlich Umlaute)
"a"	trifft auf jeden Kleinbuchstaben zu (einschließlich Umlaute und ß)
"^"	trifft auf eine Folge beliebiger Zeichen zu (auch die leere Folge). Eine sparsame Verwendung wird empfohlen, da die Bearbeitung sehr aufwendig ist.

Alle anderen Zeichen treffen nur auf ein gleiches Zeichen zu.

PROC eindeutige felder (INT CONST anzahl)

Gibt an, die wieviel ersten Felder einen Satz eindeutig identifizieren sollen. Ein Satz, der mit einem Satz der Datei in diesen Feldern übereinstimmt, wird nicht getragen. Ohne diese Angabe wird keine derartige Prüfung vorgenommen.

PROC hole satz (TEXT CONST dateiname)

Holt den letzten Satz der angegebenen Datei und fügt ihn vor dem aktuellen Satz ein.

FEHLER:

* "dateiname" existiert nicht
* Datei ist keine EUDAS-Datei
  Zieldatei existiert, ist aber keine EUDAS-Datei.
* Zieldatei hat falsche Felderzahl
  Zu viele Felder in der angegebenen Datei.
* Kein Satz zum Tragen vorhanden
  Die angegebene Datei ist leer.
* keine Datei geoeffnet
  Es muß eine virtuelle Datei vorhanden sein.

23.4 VERARBEITUNG

Die ausgewählten Sätze der aktuellen Datei können nach einer Verarbeitungsvorschrift verändert oder geprüft werden. Dies geschieht durch die Prozedur 'verarbeite'. Als Parameter kann entweder ein Verarbeitungsmuster als FILE oder die Verarbeitungsfunktion direkt als Prozedur übergeben werden.

Die Vorschrift wird durch den Operator "V" realisiert.

PROC verarbeite (FILE VAR verarbeitungsmuster)

Die aktuelle Datei wird nach dem angegebenen Muster bearbeitet. Enthält die Vorschrift, die dem ELAN-Compiler übergeben wird, einen Fehler, wird der Paralleleditor aufgerufen.

FEHLER:

* keine Datei geoeffnet
  Es muß eine virtuelle Datei vorhanden sein.

PROC verarbeite (PROC verarbeitungsfunktion)

Wie oben, nur wird die Vorschrift direkt als Prozedur übergeben.

FEHLER:

* keine Datei geoeffnet
  Es muß eine virtuelle Datei vorhanden sein.

OP V (TEXT CONST feldname, ausdruck)

Das angegebene Feld des aktuellen Satzes wird durch den Ausdruck ersetzt.

FEHLER:

* Das Feld "feldname" ist nicht definiert.
  Das angegebene Feld ist nicht vorhanden.

23.5 FUNKTIONEN IN AUSDRÜCKEN

Für Ausdrücke bei den in diesem Kapitel beschriebenen Prozeduren sind einfache Funktionen zur Abfrage von Feldinhalten vorhanden. Mit 'f' kann der Inhalt eines benannten Feldes erfragt werden, bei 'wert' wird der Inhalt erst in eine REAL-Zahl umgewandelt, wobei nichtnumerische Zeichen ignoriert werden.

Die Prozedur 'textdarstellung' kann dazu verwendet werden, den Wert einer TEXT-Variablen als TEXT-Denoter in ELAN-Syntax darzustellen.

TEXT PROC f (TEXT CONST feldname)

Liefert den Inhalt des angegebenen Feldes.

FEHLER:

* Das Feld "feldname" ist nicht definiert.

REAL PROC wert (TEXT CONST feldname)

Liefert den Inhalt des angegebenen Feldes als REAL. Dabei werden nichtnumerische Zeichen ignoriert, ausgenommen das Minuszeichen und das eingestellte Dezimalkomma (s. 'dezimalkomma'). Tritt kein numerisches Zeichen auf, wird der Wert 0.0 geliefert.

FEHLER:

* Das Feld "feldname" ist nicht definiert.

TEXT PROC textdarstellung (TEXT CONST anzeigetext)

Liefert 'anzeigetext' als TEXT-Denoter, also in Anführungsstrichen. Anführungsstriche im Text werden dabei verdoppelt. Steuerzeichen von 0 bis 31 werden in lesbare Form gebracht.

24
Anzeige

24.1 FENSTERVERWALTER

Funktionen, die einen Teil des Bildschirms in einem rechteckigen Fenster beschreiben, werden über den Fensterverwalter untereinander koordiniert. Jede Funktion fordert für ihren Fensterbereich eine Variable vom Typ FENSTER an. Vor jedem Bildschirmzugriff kann die Funktion erfahren, ob andere Programme den Bildschirm im Fensterbereich überschrieben haben. Gleichzeitig meldet sie damit Veränderungen an anderen Fenstern an, die sich mit dem eigenen überschneiden.

```
PROC fenster initialisieren (FENSTER VAR neu)
```

Jede Fenstervariable muß vor Benutzung initialisiert werden.

```
PROC fenstergroesse setzen (FENSTER VAR fenster,
                            INT CONST x anf, y anf,
                                      x laenge,
                                      y laenge)
```

Die Fenstergröße des Fensters wird gesetzt. 'x anf' und 'y anf' werden von 1..n gezählt. Die Größe eines 24x80-Bildschirms entspricht den Angaben (1, 1, 79, 24). Da das letzte Zeichen einer Zeile wegen Rollgefahr nicht benutzt werden kann, werden nur 79 Spalten angegeben.

FEHLER:

* zu viele Fenster
  Es sind nur 16 verschiedene Fenstergrößen möglich.

```
PROC fenstergroesse (FENSTER CONST fenster,
                     INT VAR x anf, y anf,
                             x laenge, y laenge)
```

Meldet die eingestellte Größe des Fensters.

```
PROC fensterzugriff (FENSTER CONST mein fenster,
                     BOOL VAR veraendert)
```

Ein Zugriff auf 'mein fenster' wird angemeldet. 'veraendert' gibt an, ob das Fenster seit dem letzten Zugriff durch einen überschneidenden Zugriff verändert wurde. Beim ersten Zugriff ist 'veraendert' immer TRUE.

PROC fenster veraendert (FENSTER CONST fenster)

Falls ein Unterprogramm eine FENSTER-Variable des Hauptprogrammes benutzt, kennzeichnet das Unterprogramm das Fenster mit dieser Prozedur als benutzt, damit das Hauptprogramm das Bild neu ausgibt.

PROC bildschirm neu

Gibt an, daß der Bildschirm von einer Funktion benutzt wurde, die ihre Zugriffe nicht über den Fensterverwalter anmeldet. Alle Fenster werden als verändert gekennzeichnet.

24.2 ANZEIGEGRUNDFUNKTIONEN

Sämtliche Anzeigefunktionen werden in einem Fenster abgewickelt, dessen Größe durch 'anzeigefenster' bestimmt wird. In diesem Fenster kann entweder ein einzelner Satz dargestellt werden, wobei jedes Feld eine Zeile einnimmt; oder es wird als Übersicht pro Zeile ein Satz angezeigt. Der entsprechende Modus wird durch 'uebersichtsanzeige' ausgewählt.

Die Funktion 'bildausgeben' übernimmt die eigentliche Ausgabe. Dabei kann durch Parameter mitgeteilt werden, ob sich an der Datei außer der Markierung etwas geändert hat. Hat sich nichts geändert, wird zur Optimierung unter Umständen nur die Markierung neu ausgegeben. Das Bild wird jedoch auf jeden Fall ganz ausgegeben, wenn das Fenster von anderer Seite verändert wurde. Auch das Öffnen einer neuen Datei wird automatisch erkannt und richtig behandelt.

Welche Felder dargestellt werden sollen, kann durch 'feldauswahl' angegeben werden. Dabei ist für jeden Anzeigemodus eine eigene Feldauswahl möglich. Die Darstellung kann durch 'rollen' in vertikaler Richtung verschoben werden.

PROC anzeigefenster (INT CONST x anf, y anf,
x laenge, y laenge)

Das Anzeigefenster wird in der entsprechenden Größe reserviert.

FEHLER:

* Anzeigefenster zu klein
  Das Fenster ist zu schmal (< 40 Zeichen), um eine sinnvolle Anzeige zuzulassen.

PROC uebersichtsanzeige (BOOL CONST ja)
Wählt Übersichts- oder Einzelsatzanzeige aus.

BOOL PROC uebersichtsanzeige
Gibt an, ob Übersichtsanzeige eingestellt ist.

PROC bild ausgeben (BOOL CONST datei veraendert)
Im Anzeigefenster wird das Bild je nach eingestelltem Modus ausgegeben, wenn das Fenster verändert wurde oder 'satz veraendert' TRUE ist. 'satz veraendert' muß immer dann angegeben werden, wenn am Inhalt der virtuellen Datei etwas verändert wurde.

PROC feldauswahl (TEXT CONST feldcode)
Die im aktuellen Modus anzuzeigenden Felder und ihre Reihenfolge werden ausgewählt. Dabei enthält 'feldcodes' an der i-ten Stelle den Code der Feldnummer des Feldes, das an i-ter Position erscheinen soll.

PROC rollen (INT CONST anzahl)
Die Darstellung wird um die angegebene Anzahl von Zeilen gerollt. Bei einer positiven Angabe wird zu höheren Feld- bzw. Satznummern gerollt (Bild bewegt sich umgekehrt). Beim ersten bzw. letzten Feld bzw. Satz hört das Rollen automatisch auf.

24.3 EDITORFUNKTIONEN

Es stehen drei Funktionen zur Verfügung, die den Editor im Anzeigemodus benutzen. Sie dienen zum Einfügen und Ändern sowie zum Eingeben eines Suckmusters. Beim Aufruf dieser Funktionen wird auomatisch zeitweise in die Einzelsatzdarstellung umgeschaltet.

Der Editor wird durch ESC 'q' verlassen. Weitere ESC Funktionen, die zum Verlassen führen sollen, können durch 'exit zeichen' angegegeben und nach Funktionsausführung mit 'exit durch' abgefragt werden.

PROC aendern
Bietet den aktuellen Satz zum Ändern an. Steht die virtuelle Datei am Ende, wird automatisch 'einfuegen' durchgeführt.

PROC einfuegen

Fügt vor dem aktuellen Satz einen Satz ein, dessen Inhalt im Editor angegeben wird.

PROC suchen

Im Editor wird eine neue Suchbedingung eingegeben.

PROC exit durch (TEXT CONST zeichenkette)

Gibt die Zeichen an, die beim Drücken nach ESC zum Verlassen des Editors führen sollen. Die eingegebenen Daten werden jedoch vorher auf jeden Fall noch verarbeitet.

TEXT PROC exit durch

Gibt an, durch welches Zeichen der Editor verlassen wurde.

25
Programmierung der Menüs

25.1 MENÜFORMAT

EUDAS verwendet drei verschiedene Strukturen zur Benutzerunterstützung: Menü, Auswahl und Hilfestellung. Ein Menü dient zur Auswahl von Funktionen, eine Auswahl bietet Feld- oder Dateinamen an und eine Hilfestellung zeigt einen Informationstext.

Alle diese Strukturen werden aus einer Initialisierungsdatei eingelesen. Die Initialisierungsdatei ist eine normale Textdatei. Ihr Format soll in diesem Abschnitt beschrieben werden.

Die Strkturen können in beliebiger Reihenfolge in der Initialisierungsdatei stehen. Jede Struktur wird durch eine spezielle Anweisung eingeleitet. Anweisungen beginnen ähnlich wie im Druckmuster mit einem Prozentzeichen. Dementsprechend gibt es die drei Anweisungen

```
% MENUE "Name"
% AUSWAHL "Name"
% HILFE "Gebiet/Name"
```

die jeweils eine Struktur einleiten. Beendet wird eine Definition immer mit

```
% ENDE
```

Für ein Menü wird noch der Text angegeben, der auf dem Bildschirm erscheinen soll. Er wird durch die Anweisung

```
% BILD
```

eingeleitet. Danach folgen Zeilen mit dem Bildschirminhalt in der gewünschten Größe (die tatsächliche Anzeigegröße wird erst beim Aufruf angegeben). Dabei werden die Auswahlpositionen, auf denen der Cursor sich bewegen kann, durch einen Trennstrich (ESC '-') dargestellt. Bei der Wahl der Positionen

ist zu beachten, daß der Cursor immer nur auf eine Position in der gleichen Zeile oder der gleichen Spalte springen kann. Eine Position, die keine weitere Position in der Zeile oder Spalte hat, kann also vom Cursor nicht erreicht werden.

Nach der Angabe des Bildes muß für jede mögliche Auswahlposition noch eine weitere Angabe gemacht werden. Die Auswahlpositionen (oder Felder) werden dabei von links nach rechts und von oben nach unten durchgezählt. Begonnen wird mit der Nummer 1.

Eine Felddefinition hat das Format

```
% FELD nr "Hilfstext" "Tasten"
```

Die Nummer identifiziert das Feld. Der Hilfstext gibt den Namen der Hilfestellung an, die gezeigt werden soll, wenn auf diesem Feld ESC '?' gedrückt wird. Die Tasten sind alle Zeichen, die gedrückt werden können, um dieses Feld direkt auszuführen.

Anschließend an die Felddefinition kann noch ein ELAN-Programm folgen, daß bei Auswahl des Feldes ausgeführt wird.

Zum Schluß können noch Ersetzungen angegeben werden. Diese dienen dazu, zur Laufzeit wechselnde Informationen in das Menübild einzublenden (z.B. Datum und Uhrzeit). Die Ersetzungsdefinitionen werden durch die Anweisung

```
% ERSETZE
```

eingeleitet. Eine einzelne Ersetzungsdefinition hat dann folgenden Aufbau

```
&IDENT nr "Fülltext"
```

Die Bezeichnung am Anfang der Zeile gibt den Text an, der im Bild ersetzt wird. Dieser Text muß auch irgendwo so in der Bilddefinition auftauchen. Die Nummer gibt an, durch welchen Text dieses Muster ersetzt wird. Diese Nummer ist dabei durch das Programm festgelegt, das die Information unter dieser Nummer ablegt. Der Fülltext gibt an, womit aufgefüllt werden soll, wenn der Ersetzungstext kürzer ist als das Muster. Gleichzeitig begrenzt er auch die maximale Länge der Ersetzung. Wird hier ein leerer Text angegeben, so kann die Ersetzung beliebig lang sein.

Für eine Auswahl muß zuerst ein Vorspann angegeben werden, der immer in den ersten Bildschirmzeilen der Auswahl angezeigt wird. Dieser wird durch

```
% VORSPANN
```

eingeleitet. Danach folgt das Bild. Das Bild setzt sich aus drei Teilen zusammen. Die erste Zeile, in der ein Trennstrich vorkommt, bildet den Wiederholungsteil. Diese Zeile wird nachher so oft wie nötig mit entsprechenden Inhalten wiederholt. Die Teile davor und danach werden jeweils bei Bedarf mitgerollt. Hier können ebenso wie im Vorspann Ersetzungen angegeben werden.

Die Wiederholungszeile darf mehrere Trennstriche enthalten. Die Inhalte werden dann in mehreren Spalten angezeigt. Zu beachten ist, daß vor einem Trennstrich noch mindestens fünf Zeichen Platz für eine laufende Nummer bleiben müssen.

Auch bei einer Auswahl kann zum Schluß ein Ersetzungsteil angegeben werden.

Der Name einer Hilfestellung setzt sich aus zwei Teilen zusammen, die durch einen Schrägstrich getrennt werden. Der erste Name gibt die Kategorie der Hilfestellung an, der zweite Name den Einzeltext. Dies dient dazu, die relativ große Zahl der Hilfstexte überschaubar zu machen. Als Beispiel

```
% HILFE "JA/Allgemein"
```

Eine Hilfestellung besteht einfach aus mehreren Seiten Text. Jede Seite wird durch die Anweisung

```
% SEITE nr
```

eingeleitet. Die Nummer hat nur Kommentarcharakter. Anschließend folgen alle Zeilen der angegebenen Seite. Es kann jedoch auch eine Seite einer vorher definierten Hilfestellung angegeben werden. Dadurch brauchen gleiche Seiten nicht mehrfach gespeichert werden. Dies geschieht in der Form

```
% SEITE nr "Hilfsname"
```

Die Nummer bezieht sich jetzt auf die entsprechende Seite der angegebenen Hilfestellung, wobei die Seiten einfach von 1 an durchgezählt werden. Anschließend darf für diese Seite natürlich kein Text folgen.

25.2 VERWALTUNG DER MENÜS

Alle Menüdaten werden intern in Datenräumen gehalten. Dabei sind die einzelnen Strukturen getrennt. Mit 'menuedaten einlesen' können Menüdaten aus einer Struktur gelesen und abgespeichert werden. Existierende Strukturen werden dabei überschrieben, neue hinzugefügt.

Mit 'menuenamen' können die vorhandenen Strukturen abgefragt werden. Mit 'menue loeschen' werden einzelne Strukturen oder alle Menüdaten gelöscht. Damit die Datenräume mit den Menüdaten auch an Söhne übergeben werden, ist der 'global manager' entsprechend geändert. Dies wird im einzelnen durch 'menue manager' bewirkt. Der neue Global Manager akzeptiert auch Anfragen von Tasks, die nicht Söhne oder Enkel sind.

`PROC menuedaten einlesen (FILE VAR f)`

Die in der Datei enthaltenen Strukturen werden eingelesen und abgespeichert. Treten bei diesem Prozeß Fehler auf, so wird eine entsprechende Meldung ins EUMEL-Notizbuch geschrieben.

`THESAURUS PROC menuenamen (INT CONST index)`

Liefert die Namen der Strukturen. Der Index hat folgende Bedeutung:

1: Hilfskategorien
2: Menüs
3: Auswahlen

Ist der Index negativ, so werden die Hilfsnamen der entsprechenden Hilfskategorie geliefert, die im Thesaurus den Index mit umgekehrtem Vorzeichen hat.

`PROC menue loeschen (TEXT CONST name, INT CONST index)`

Löscht den Namen in dem Thesaurus mit dem angegebenen Index, falls dort der Name vorhanden ist.

`PROC menue loeschen`

Löscht alle Menüdaten.

PROC global manager

Geänderter Manager, der die Menüdatenräume in Söhne transportiert. Nur im Multi-User-System vorhanden.

```
PROC menue manager (DATASPACE VAR ds,
                    INT CONST order, phase,
                    TASK CONST order task)
```

Eigentliche Manager-Routine. Kann dazu dienen, Managererweiterungen vorzunehmen. Nur im Multi-User-System vorhanden. Beispiel: der Manager soll nur Aufträge von Söhnen annehmen.

```
PROC new manager (DATASPACE VAR ds,
                  INT CONST order, phase,
                  TASK CONST order task):

  LET begin code = 4;
  IF order task < myself OR order = begin code OR
     order task = supervisor THEN
    menue manager (ds, order, phase, order task)
  ELSE
    errorstop ("kein Sohn")
  END IF

END PROC new manager;

global manager (PROC new manager)
    (* startet den Manager *)
```

25.3 AUFRUF

Menüs werden mit der Prozedur 'menue anbieten' aufgerufen. Dabei muß neben dem Namen ein Fenster übergeben werden, in dem das Menü dann angezeigt wird. Es ist darauf zu achten, daß das Fenster groß genug ist, um wenigstens die Auswahlpositionen im Bild zu haben.

Außerdem muß eine Prozedur übergeben werden, die die einzelnen Funktionen ausführt. Diese Prozedur erhält als Parameter die Nummer der ausgewählten Funktion.

Sind im Menübild Ersetzungen angegeben, so können die gewünschten Texte vor dem Aufruf mit 'ersetzungstext' angegeben werden.

Eine Auswahl wird mit der Prozedur 'auswahl anbieten' aufgerufen. Diese bekommt ebenfalls ein Fenster übergeben, außerdem den Namen einer Hilfestellung, die bei Bedarf aufgerufen werden kann. Weiterhin muß eine Prozedur übergeben

werden, die die auszuwählenden Texte erzeugt. Die gewählten Texte können anschließend mit der Prozedur 'wahl' abgefragt werden.

Eine Hilfestellung wird durch die Prozedur 'hilfe anbieten' ausgegeben. Dabei muß außer dem Namen auch noch ein Fenster übergeben werden.

Zur Hilfestellung gehört auch die Statuszeile. Mit 'statuszeile' kann festgelegt werden, wo und wie die Statuszeile erscheinen soll. Mit 'status anzeigen' kann ein Text in der Statuszeile ausgegeben werden. Die Menüprogramme tun dies für Ihren Bereich jedoch selbst.

```
PROC menue anbieten (TEXT CONST name, FENSTER VAR f,
                     BOOL CONST esc erlaubt,
                     PROC (INT CONST) interpreter)
```

Das angegebene Menü wird in dem Fenster 'f' angezeigt. Das Fenster wird nach Beendigung des Menüs als verändert gekennzeichnet. 'esc erlaubt' gibt an, ob mit ESC ESC ein Kommandodialog geführt werden kann. Die übergebene Prozedur muß die einzelnen Funktionen ausführen, die als Zahl übergeben werden. Der Interpreter wird im 'disable stop' aufgerufen, daher ist in der Regel ein 'enable stop' erforderlich.

FEHLER:

* "Name" existiert nicht.
  Das angegebene Menü ist nicht vorhanden.

```
PROC ersetzungstext (INT CONST nr, TEXT CONST text)
```

Es sind Nummern von 4 bis 9 möglich, die mit beliebigen Texten belegt werden können. Die Nummern 1 bis 3 werden vom System selbst erzeugt, und zwar:

1: Uhrzeit (hh:mm)
2: Datum (tt.mm.jj)
3: Taskname

```
PROC auswahl anbieten (TEXT CONST name, FENSTER CONST f,
                       TEXT CONST hilfe,
                       PROC (TEXT VAR, INT CONST) inhalt)
```

Ruft die Auswahl in dem angegebenen Fenster auf. 'hilfe' wird als Hilfestellung verwendet. Die Prozedur 'inhalt' liefert den Text, der an der n-ten Position stehen soll. Wenn keine weiteren Texte vorhanden sind, muß für alle weiteren Positionen "" geliefert werden.

FEHLER:
* "Name" existiert nicht.
Die angegebene Auswahl ist nicht vorhanden.

INT PROC wahl (INT CONST stelle)
Liefert die Nummer des Textes, der an der n-ten Stelle ausgewählt wurde. Wurde kein weiterer Text mehr gewählt, so wird 0 geliefert.

PROC hilfe anbieten (TEXT CONST name, FENSTER CONST f)
Bietet den Hilfstext in dem angegebenen Fenster an. Existiert die Hilfe innerhalb der angegebenen Kategorie nicht, wird die erste Hilfe der Kategorie angezeigt.
FEHLER:
* Hilfe existiert nicht
Die angegebene Hilfskategorie ist nicht vorhanden.

PROC statuszeile (TEXT CONST vorher, nachher)
Gibt die Texte an, die vor und nach der Statuszeile ausgegeben werden sollen. Der Text vorher sollte in der Regel eine Cursorpositionierung durchführen, der Text nachher bis zum Ende der Zeile löschen. Wenn als Text vorher "" angegeben ist, wird keine Statuszeile ausgegeben.

PROC status anzeigen (TEXT CONST zeile)
Zeigt den angegebenen Text in der Statuszeile an.

25.4 DIALOG

EUDAS führt den Benutzerdialog in einem Fenster, dessen Größe durch 'dialogfenster' eingestellt wird. Das Dialogfenster wird zeilenweise belegt. 'dialog' reserviert eine neue Zeile für eine beliebige Ausgabe. Wenn das Fenster voll ist oder überschrieben wurde, wird in der ersten Zeile wieder angefangen. Den gleichen Effekt kann man mit 'neuer dialog' erreichen.

Mit 'ja' kann man im Dialogfenster eine Frage stellen. Als Antwort kann auch eine Hilfestellung angefordert werden oder die Funktion abgebrochen werden. Mit 'editget' kann man nach Aufforderung einen Text eingeben lassen. Dabei stehen die gleichen zusätzlichen Möglichkeiten zur Verfügung.

Mit 'fehler ausgeben' kann eine durch 'errorstop' erzeugte Fehlermeldung im Dialogfenster angezeigt werden. Der Benutzer muß erst eine Taste drücken, ehe er weitermachen kann. Auch hier kann eine Hilfestellung zu der Fehlermeldung abgerufen werden.

```
PROC dialogfenster (INT CONST x, y, x1, y1)
```

Gibt die Fenstergröße und -position des Dialogfensters an. Für den ganzen Bildschirm müßte (1, 1, 79, 24) angegeben werden.

```
PROC dialog
```

Positioniert den Cursor auf die nächste Dialogzeile im Dialogfenster. Wurde das Fenster verändert, wird das Fenster gelöscht und auf die erste Zeile positioniert.

```
PROC neuer dialog
```

Sorgt dafür, daß der nächste Aufruf von 'dialog' wieder in einem leeren Fenster beginnt.

```
BOOL PROC ja (TEXT CONST frage, hilfe)
```

Stellt die angegebene Frage. Es kann die angegebene Hilfsinformation abgerufen werden. Außerdem wird durch ESC 'a' ein Abbruch (errorstop ("")) erzeugt.

```
PROC editget (TEXT CONST prompt, TEXT VAR eingabe,
              TEXT CONST res, hilfe)
```

Gibt den Text 'prompt' aus und editiert dahinter 'eingabe'. Es kann die angegebene Hilfsinformation abgerufen werden. 'res' gibt an, bei welchen ESC-Folgetasten das Editieren beendet werden soll. In einem solchen Fall wird als Eingabe (ESC + die gedrückte Taste) zurückgeliefert.

```
PROC fehler ausgeben
```

Im Dialogfenster wird die letzte Fehlermeldung ausgegeben. Gleichzeitig wird der Fehlerzustand gelöscht. Der Benutzer muß eine Taste drücken, um weiterzumachen. Alternativ kann die Hilfsinformation "FEHLER/" + text (errorcode) abgerufen werden.

ANHANG

Anhang A
Fehlermeldungen

In diesem Kapitel sind alle Fehlermeldungen aufgeführt, die von EUDAS erzeugt werden und zum Abbruch einer Funktion führen können.

illegale Feldnummer
Es wurde eine Nummer als Feldnummer angegeben, die nicht erlaubt ist.

Datei ist keine EUDAS-Datei
Es wurde versucht, eine andere Datei als EUDAS-Datei zu bearbeiten.

inkonsistente EUDAS-Datei
Die interne Struktur der Datei ist zerstört. Kann durch Hardwareprobleme (Archiv-Lesefehler) oder EUDAS-interne Fehler entstanden sein.

EUDAS-Datei voll
Eine EUDAS-Datei kann nur eine bestimmte Anzahl von Sätzen aufnehmen (mindestens 5000).

Nicht erlaubtes Dezimalkomma
Als Dezimalkomma kann nur ein einzelnes Zeichen angegeben werden.

Zuviel Dateien geoeffnet
Es können nicht mehr als 10 Dateien gleichzeitig geöffnet, gekettet und gekoppelt sein.

Zu viele Felder
Alle geöffneten Dateien zusammen dürfen nicht mehr als 256 Felder der virtuellen Datei ergeben.

Zu viele Koppelfelder
Es dürfen insgesamt nicht mehr als 32 Koppelfelder entstehen.

keine Koppelfelder vorhanden
Eine Datei kann nicht gekoppelt werden, wenn Sie kein Koppelfeld besitzt.

keine Datei geoeffnet
Es kann nicht gekettet oder gekoppelt werden, wenn noch keine Datei geöffnet wurde. Ebenfalls sind keine Verarbeitungsprozeduren möglich.

Datei nicht gesichert
Eine vorher geöffnete Datei ist verändert und nicht gesichert.

Suchmuster zu umfangreich
Ein Suchmuster darf nicht mehr als 100 Vergleiche erfordern.

direkt Drucken nicht moeglich
Entweder ist kein Druckprogramm installiert oder die Spooltask reagiert nicht.

Das Feld "Feldname" ist nicht definiert
Sie haben einen falschen Namen angegeben.

Kein Satz zum Tragen vorhanden
Es wurde versucht, aus einer leeren Datei oder am Dateiende zu tragen.

Zieldatei hat falsche Felderzahl
Eine Zieldatei beim Tragen hat weniger Felder als die aktuelle Datei. Daher würden beim Tragen Informationen verlorengehen.

Das Feld "Feldname" verletzt die Pruefbedingung
Eine Prüfprozedur wurde außerhalb des Tragens aufgerufen und die Bedingung war nicht erfüllt.

Das Feld "Feldname" ist nicht in der Wertemenge
Eine Prüfprozedur wurde außerhalb des Tragens aufgerufen und die Bedingung war nicht erfüllt.

Das Feld "Feldname" stimmt nicht mit der Maske ueberein
Eine Prüfprozedur wurde außerhalb des Tragens aufgerufen und die Bedingung war nicht erfüllt.

Zu viele Fenster
Es sind nicht mehr als 16 verschiedene Größen von Fenstern möglich.

Fenster zu klein
Ein Menü wurde in einem zu kleinen Fenster aufgerufen.

Hilfe existiert nicht
Es wurde versucht, eine nicht vorhandene Hilfestellung aufzurufen.

Hilfe ist leer
Die angewählte Hilfestellung enthält keinen Text.

Anzeigefenster zu klein
Das Anzeigefenster muß mindestens 40 Zeichen breit sein.

Bitte erst eine Datei oeffnen !
Das gewählte Menü kann erst dann gezeigt werden, wenn vorher eine Datei geöffnet wurde.

Ungueltige Satznummer
Der angegebene Text stellt keine Satznummer dar.

Aenderungen sind nicht erlaubt
Es wurde versucht, Änderungen an der aktuellen Datei vorzunehmen, obwohl Änderungen nicht erlaubt sind.

Anhang B

Prozeduren mit Parametern

:= (SATZ VAR, SATZ CONST) 21.1

aendern 24.3
aendern erlaubt : BOOL 22.1
aenderungen eintragen 22.4
anzahl dateien : INT 22.1
anzahl felder : INT 22.2
anzahl koppeldateien : INT 22.1
anzeigefenster (INT CONST x anf, y anf,
x laenge, y laenge) 24.2
auf satz (EUDAT VAR, INT CONST satznr) 21.3
auf satz (EUDAT VAR, TEXT CONST muster) 21.3
auf satz (INT CONST satznr) 22.3
auswahl anbieten (TEXT CONST name, FENSTER CONST f,
TEXT CONST hilfe,
PROC (TEXT VAR, INT CONST) inhalt)
25.3

bild ausgeben (BOOL CONST datei veraendert) 24.2
bildschirm neu 24.1

dateiende (EUDAT CONST) : BOOL 21.3
dateiende : BOOL 22.3
dateien loeschen (BOOL CONST auch geaenderte) 22.1
dateiversion : INT 22.1
dezimalkomma (TEXT CONST komma) 21.5
dezimalkomma : TEXT 21.5
dialog 25.4
dialogfenster (INT CONST x anf, y anf,
x laenge, y laenge) 25.4
direkt drucken (BOOL CONST ja) 23.1
druckdatei (TEXT CONST dateiname) 23.1
drucke (TEXT CONST mustername) 23.1

editget (TEXT CONST prompt, TEXT VAR eingabe,
TEXT CONST res, hilfe) 25.4
eindeutige felder (INT CONST anzahl) 23.3
einfuegen 24.3
ersetzungstext (INT CONST nr, TEXT CONST text) 25.3
eudas dateiname (INT CONST dateinr) : TEXT 22.1
EUDAT 21.2
exit durch (TEXT CONST exit zeichen) 24.3
exit durch : TEXT 24.3

f (TEXT CONST feldname) : TEXT 23.5
fehler ausgeben 25.4
feld aendern (SATZ VAR, INT CONST feldnr,
TEXT CONST inhalt) 21.1
feld aendern (EUDAT VAR, INT CONST feldnr,
TEXT CONST inhalt) 21.4
feld aendern (INT CONST feldnr, TEXT CONST inhalt) 22.2
feldauswahl (TEXT CONST feldcodes) 24.2
feld bearbeiten (SATZ CONST, INT CONST feldnr,
PROC (TEXT CONST, INT CONST, INT CONST) bearbeite)
21.1
feld bearbeiten (EUDAT CONST, INT CONST feldnr,
PROC (TEXT CONST, INT CONST, INT CONST) bearbeite)
21.4
feld bearbeiten (INT CONST feldnr,
PROC (TEXT CONST, INT CONST, INT CONST) bearbeite)
22.2
felderzahl (SATZ CONST) : INT 21.1
felderzahl (EUDAT CONST) : INT 21.2
feldindex (SATZ CONST, TEXT CONST muster) : INT 21.1
feldinfo (EUDAT VAR, INT CONST feldnr, info) 21.5
feldinfo (EUDAT CONST, INT CONST feldnr) : INT 21.5
feld lesen (SATZ CONST, INT CONST feldnr,
TEXT VAR inhalt) 21.1
feld lesen (EUDAT CONST, INT CONST feldnr,
TEXT VAR inhalt) 21.4
feld lesen (INT CONST feldnr, TEXT VAR inhalt) 22.2
feldmaske (TEXT CONST feldname, maske) 23.3
feldnamen aendern (EUDAT VAR, SATZ CONST namen) 21.2
feldnamen bearbeiten (INT CONST feldnr,
PROC (TEXT CONST, INT CONST, INT CONST) bearbeite)
22.2
feldnamen lesen (EUDAT CONST, SATZ VAR namen) 21.2
feldnamen lesen (INT CONST feldnr, TEXT VAR name) 22.2
feldnummer (TEXT CONST feldname) : INT 22.2
FENSTER 24.1
fenstergroesse (FENSTER CONST f,
INT VAR x anf, y anf,
x laenge, y laenge) 24.1
fenstergroesse setzen (FENSTER VAR fenster,
INT CONST x anf, y anf,
x laenge, y laenge)
24.1
fenster initialisieren (FENSTER VAR fenster) 24.1
fenster veraendert (FENSTER CONST fenster) 24.1
fensterzugriff (FENSTER CONST fenster,
BOOL VAR veraendert) 24.1

global manager 25.2

hilfe anbieten (TEXT CONST name, FENSTER CONST f) 25.3
hole satz (TEXT CONST dateiname) 23.3

```
inhalt veraendert (INT CONST dateinr)   BOOL          22.1
ja (TEXT CONST frage, hilfe) : BOOL                   25.4

K (TEXT CONST feldname, ausdruck)                     23.2
kette (TEXT CONST dateiname)                          22.1
kopiere (TEXT CONST dateiname, FILE VAR muster)       23.2
kopiere (TEXT CONST dateiname, PROC kopierfunktion)   23.2
kopple (TEXT CONST dateiname)                         22.1

markierte saetze : INT                                22.5
markierung aendern                                    22.5
markierungen loeschen                                 22.5
maxdruckzeilen (INT CONST anzahl zeilen)              23.1
menue anbieten (TEXT CONST name, FENSTER VAR f,
                BOOL CONST esc erlaubt,
                PROC (INT CONST) interpreter)         25.3
menuedaten einlesen (FILE VAR f)                      25.2
menue loeschen (TEXT CONST name, INT CONST index)     25.2
menue loeschen                                        25.2
menue manager (DATASPACE VAR ds,
               INT CONST order, phase,
               TASK CONST order task)                 25.2
menuenamen (INT CONST index) : THESAURUS              25.2

neuer dialog                                          25.4
notizen aendern (EUDAT VAR, INT CONST notiz nr,
                 TEXT CONST notizen)                  21.2
notizen lesen (EUDAT CONST, INT CONST notiz nr,
               TEXT VAR notizen)                      21.2

oeffne (EUDAT VAR, TEXT CONST dateiname)              21.2
oeffne (TEXT CONST dateiname,
        BOOL CONST aendern erlaubt)                   22.1

pruefe (TEXT CONST feldname, BOOL CONST bedingung)    23.3

reorganisiere (TEXT CONST dateiname)                  21 5
rollen (INT CONST anzahl)                             24.2

saetze (EUDAT CONST) : INT                            21.3
SATZ                                                  21.1
satz aendern (EUDAT VAR, SATZ CONST neuer satz)       21.4
satz ausgewaehlt : BOOL                               22.5
satz einfuegen (EUDAT VAR, SATZ CONST satz)           21.4
satz einfuegen                                        22.4
satz initialisieren (SATZ VAR satz)                   21.1
satzkombination : INT                                 22.3
satz lesen (EUDAT CONST, SATZ VAR satz)               21.4
satz loeschen (EUDAT VAR)                             21.4
satz loeschen                                         22.4
satz markiert : BOOL                                  22.5
satznr (EUDAT CONST) : INT                            21.3
satznummer : INT                                      22.3
```

```
sichere (INT CONST dateinr, TEXT CONST dateiname)    22.1
sortiere (EUDAT VAR, TEXT CONST reihenfolge)         21.5
sortiere (EUDAT VAR)                                 21.5
sortierreihenfolge (EUDAT CONST) : TEXT              21.5
status anzeigen (TEXT CONST zeile)                   25.3
statuszeile (TEXT CONST text vorher, text nachher)   25.3
std kopiermuster (TEXT CONST dateiname. FILE VAR f)  23.2
suchbedingung (INT CONST feldnr,
               TEXT CONST bedingung)                 22.5
suchbedingung loeschen                               22.5
suchen                                               24.3

textdarstellung (TEXT CONST text) : TEXT             23.5
trage (TEXT CONST dateiname, FILE VAR protokoll,
       BOOL CONST test)                              23.3
trage satz (TEXT CONST dateiname)                    23.3

uebersichtsanzeige (BOOL CONST modus)                24.2
uebersichtsanzeige : BOOL                            24.2
unsortierte saetze (EUDAT CONST) : INT               21.5

V (TEXT CONST feldname, ausdruck)                    23.4
verarbeite (FILE VAR verarbeitungsmuster)            23.4
verarbeite (PROC verarbeitungsfunktion)              23.4

wahl (INT CONST stelle) : INT                        25.3
weiter (EUDAT VAR)                                   21.3
weiter (EUDAT VAR, TEXT CONST muster)                21.3
weiter (INT CONST modus)                             22.3
wert (TEXT CONST feldname) : REAL                    23.5
wertemenge (TEXT CONST feldname, menge)              23.3

zurueck (EUDAT VAR)                                  21.3
zurueck (EUDAT VAR, TEXT CONST muster)               21.3
zurueck (INT CONST modus)                            22.3
```

Anhang C
Register

Abbruch 105, 107, 124
Abkürzungen 79ff, 129, 131, 134
ABKUERZUNGEN 80
Abkürzungsteil 80f, 86, 88, 129
Abkürzungszeilen 129
Absatzmarkierungen 97
Abschlußzeile 105, 120
Abschnitte 38f, 88, 134
Ändern 29, 48, 69, 122
  Koppeldatei 51
Änderung, Datum der letzten 147
Änderungen 115f, 138, 157
Alternative, globale 54, 126
  lokale 53, 125
Ankoppeln 146
Ankreuzen 18, 105
Anweisungen 35, 87, 128
Anzeige 63ff, 119ff, 170
Anzeigeformat 119
Arbeitskopie 31
Archiv 17f, 109
Archivname 109f
Archivübersicht 17f, 109
Attribut 2, 16
Ausgabedatei 97, 130
Auswahl 18, 105, 173

Bearbeiten 76
Bearbeitungsvorschrift 76
Bedienung 11ff.
Bewegen 22, 24, 51, 115, 119
Bildausgabe 121
Bildschirm, Aufbau 21
  Ausschnitt 63
blättern 23

Cursor 12
Cursorbewegung 103, 106
Cursortasten 103

date 83
Datei 15
  ansehen 21
  einrichten 27, 57
  löschen 19, 110
  Platzbedarf 62
  reorganisieren 111, 138
  umbenennen 111
  umstrukturieren 75
  virtuelle 48, 113
Dateiauswahl 18, 48, 114, 127
Dateiende 119
Dateiname 15
  Eingabe 20
Dateistruktur, beim Ketten 113
Dateiübersicht 19, 110
Dateiverwaltung 17, 109ff
Datenraum 146
Datenverwaltung 1
Datum 79
DATUM 61, 138
Dezimalkomma 61, 137, 150
Dezimalpunkt 61
Dialog 103, 107, 179
DIN 61, 137
direkt drucken 36
Doppeleinträge 71
Druckausgabe 161
druckdatei 97
Druckdatei 34, 128
Drucken 5, 33ff, 127ff, 161
  Ablauf 37
Drucker 36

Druckersteuerungsanweisungen 44, 98
Druckgenerator 36
Druckmuster 5, 33, 127ff
  Dateilimit 96, 133
  erstellen 36
  Fehler 37, 81, 134ff
  Übersetzung 88
Druckvorgang 129f

Editieren, Zeile 107
Editor 16, 20, 36, 123, 171
eindeutige felder 141
Einfügen 28, 48, 122
Einzelbedingungen 125
ELAN 6
ELAN-Ausdruck 74, 77f, 80, 92, 139
ELAN-Compiler 76, 81, 128, 134f
Endesatz 22
Ersetzungen 174
eudas 11, 108
EUDAS-Archiv 7
EUDAS-Datei 15ff
  Beschränkungen 17
  Struktur 16

Fehler 105
Feld 16, 147
Feldauswahl 64, 65, 122
Felder 145
Feldinhalt 16, 120
Feldmaske 70, 141
Feldmuster 35, 39ff, 81, 131
  Länge 132
  Position 131
Feldname 16, 35, 114, 120, 137, 139
  Länge 79, 121
  Eingabe 27
Feldreihenfolge 75
Feldstruktur 48, 61, 68, 137, 154
Feldtypen 60, 137
Fenster 169
Formbriefe 43
Formular 6, 33
Fortsetzungszeilen 99
Frage 107
Funktion ausführen 104

Generierung 8, 9
Global Manager 176
Gruppen 91ff, 97, 129
Gruppennummer 91
Gruppenwechsel 130

Hauptdatei 113
Hauptmenü 11, 14, 108
Hilfestellung 13, 103, 107, 173
Hintergrunddiskette 8

IF-Anweisungen 78, 84f
Initialisierungsteil 90, 91, 98, 128, 134
Installation 7

K 74, 139
Karteikarte 3, 22
Karteikasten 4
Ketten 47, 113, 152
Klammern, spitze 42
Kombination 58, 115, 155
Kombinationsnummer 51
Kommandodialog 104
Kommandozeilen 128
Kopieren 73ff, 139, 163
Kopiermuster 73, 139
Koppelfeld 49, 114
Koppeln 48f, 114, 152
Koppelsatz 115
Koppelverfahren 50

Leertaste 12
Lernmodus 30, 103
lfd nr 83
Lieferumfang 7
lineform 97

list 110

Markieren 57f, 158
Maske 70
maxdruckzeilen 98
MEHR 100, 134
Menü 11ff., 103ff, 173
  Struktur 13, 108
  Tiefe 14
Menüprinzipien 11
Modi 98ff, 133
Monitor 109
Multi-User 8
Musterbrief 85
Musterteil 81, 129
  bedingter 87
Musterzeichen 40, 95, 131
Musterzeile 35, 129
  Interpretation 131ff
Muttersystem 8

Nachspann 38, 92, 130
Namenskonflikte 115
Normalmodus 99
Notizen 147

Objekt, Eigenschaften 2
ODER-Verknüpfung 125
Öffnen 19, 27, 113, 151

pageform 97
Paralleleditor 37, 89
Pfeiltasten 12
Positionierung 156
Priorität 126
Programmzeilen 129
Proportionalschrift 97
Prüfbedingungen 68f, 137, 140, 147, 165
pruefe 140
PUBLIC 9
Pufferplatz 132

Refinement 76, 80, 86, 88, 92, 129, 134
Reihenfolge 105
Reorganisieren 62, 138
RETURN 20
Rollen 63, 65, 105, 121, 124
RUBIN 20, 30
RUBOUT 20, 29

Satz 16, 145, 147
  ändern 29, 69
  einfügen 28, 116, 117
  holen 68, 141
  Inhalt 22
  löschen 67, 115, 117
  tragen 67, 141
Satzeditor 29, 63, 65, 123f
Satzkombination 120
Satznummer 21, 120, 122
Satzposition 147, 155
save system 8, 9
Schreibmarke 12
Schrifttypen 44
Seitenformatierung 97
Selektion 23, 119, 124
  löschen 25
Sicherheitskopie 31, 114, 117
Sichern 31, 48, 117, 153
Single-User 8
Sortieren 4, 59ff, 137, 138, 149
Sortierreihenfolge 59, 138
Sortierzustand 138
Spaltendruck 96, 130
Standard-Kopiermuster 73, 139
Standardformular 64, 119
Statuszeile 11, 19, 103, 107
Stern 55
SUCH 24
Suchbedingung 24, 158
Suchen 4, 23, 122, 137
  Geschwindigkeit 57
  Optimierung 126

Suchmuster 26f, 38, 53ff, 124ff
  Eingabe 23, 29

Tabellenmodus 98
TEXT 61, 137
Textdatei 15, 33, 77, 127
  erstellen 36
Textkonstanten 85
Textkosmetik 44, 97
Textverarbeitung 1
Textzeilen 128
Tragen 67ff, 140, 164

Übersichtsanzeige 64, 121
Uhrzeit 79
Umlaute 90
UND-Verknüpfung 125

V 77, 142
Verarbeitung 167
Verarbeitungsvorschrift 141
Verketten 47
Verneinung 56, 126
Vorspann 38, 92, 130

wert 95
Wertemenge 69, 140
WIEDERHOLUNG 35
Wiederholungsteil 129

ZAHL 60, 137
Zeile einfügen 30, 124
Zeile löschen 30, 124
Zeilenende 133
Zeilentypen 128f
Zeilenwiederholung 99, 133